電腦時代

COMPUTING AGE 24

工商業●教育●家庭娛樂●私人應用電腦雜誌　港幣十元

隨「機」應變　確保資料

PROMAS POWER ESCORT

自動後備電源供應器

電源中斷，電壓驟降等意外，令閣下痛失寶貴資料或損壞昂貴精密電腦器材，損失無可挽救！

精明的電腦用家，事事計劃周全，自當配備PROMAS自動後備供電器，以備電源故障時，隨時提供足夠電源，確保電腦運作可順利完成。

- 可用作防止電壓下降，電壓沖擊及電源中斷等，確保電源供應正常。
- 當電力停止供應時，能在500分之一秒內自動提供後備電源，並以蜂鳴器報警及指示燈顯示。
- 內置過荷保護線路及多種過濾器，提供高質素電源。
- 供電時間較一般自動後備供電器為長。
- 在電力供應正常後，亦可隨即自動充電。
- 外型簡單美觀，適用於任何電腦硬件儲存系統、精密電子儀器及一般電器使用。
- 不同型號，多種功率，可供選擇（PE 250、PE 501及PE 1000 A）

PROMAS COMPUTER CENTRE LTD.

寶馬士電腦有限公司

[illegible]22–26號[illegible]608室　電話：3-674687　電訊：HX 38316

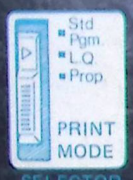
Super 5 EP-1201

(New version of EN-P109[illegible]

A TWO-IN-ONE PRINTER

IDEAL FOR PERSONAL & BUSINESS USE

- PERFECT LETTER QUALITY (PLQ)
- 80/132 COLUMN EASILY SWITCHIBLE
- IBM COMPATIABLE PC/XT
- IBM GRAPHIC MODE
  DIP SW SELECT
- PRINT MODE SELECTOR SWITCH



SELECTOR SWITCH

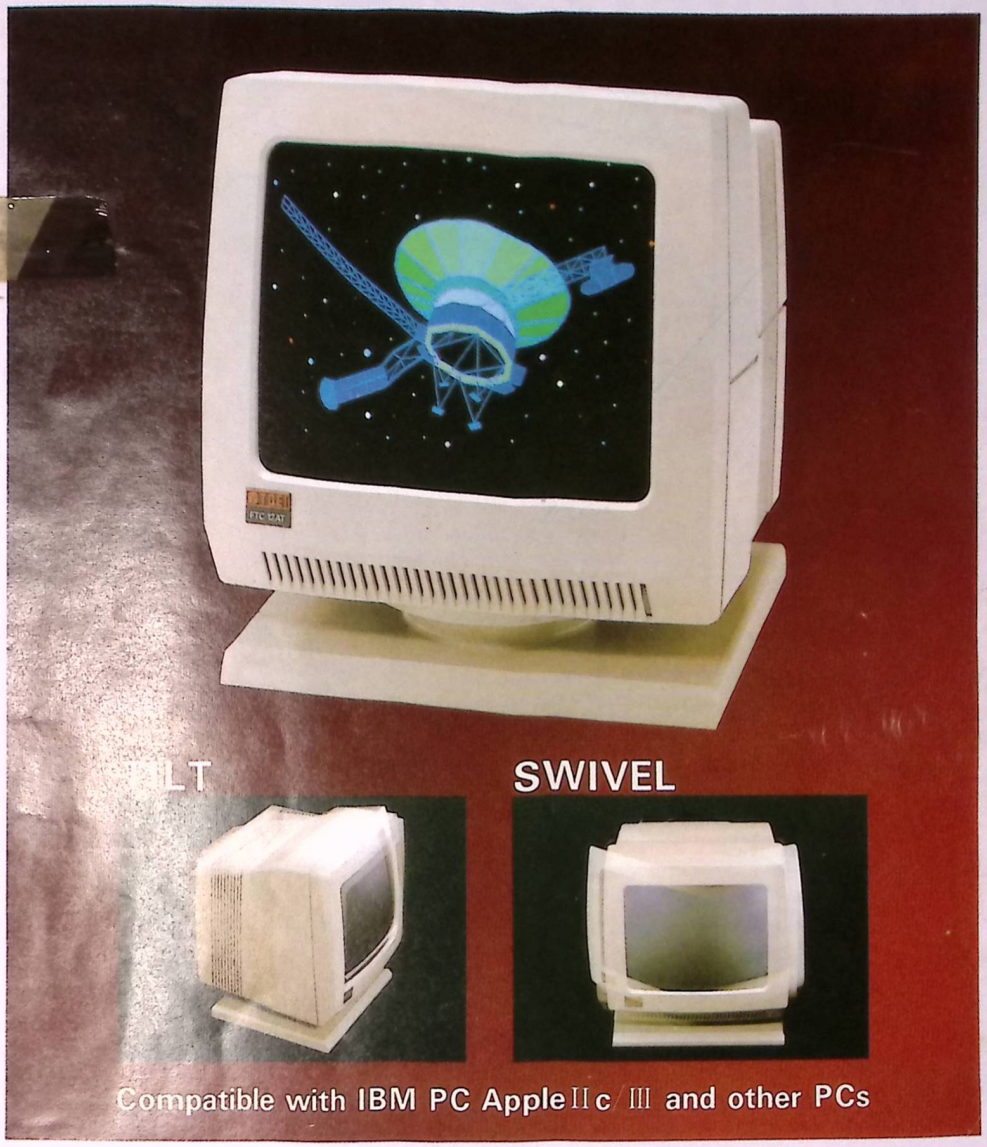
EN-P 1090

Super 5

A & N ENTERPRISE CO., LI ..O3
20 F WESTERN CENTRE
40 50 DES VOEUX ROAD WEST, H H T23
TEL 5-461179 5-488463
TELEX: 66218 MCLWW HX

TOEI 系列彩色監視器

色彩更趨完美・影像更見逼眞



FTC系統規格表

Name of Model		FTC-12AT	FTC-12AT-HM	FTC-12AT-HF	FTC-12AT-W
Picture Tube		12", 90°Deflection, In-Line Type, P22, Hi-Contrast CRT (FTC-12AT, FTC-12AT-HM : Non-Glare)			
		0.38㎜ Dot Trio Pitch		0.28㎜ Dot Trio Pitch	
Input Signal Level	RGB Video Signal	RGB Analog 2~5Vpp (Positive), RGB plus Intensity TTL Level (Positive)	RGB plus Intensity TTL Level (Positive)		
	Sync. Signal	Composite or Separate (H. D., V.D.) TTL Level (Positive or Negative)			
Scanning Frequency		H. 15.75kHz ± 400Hz V. 60Hz (50~65Hz)	H. 25kHz ± 600Hz, V. 60Hz (50~65Hz)		H. 31.5kHz ± 1kHz V. 60Hz (50~65Hz)
Video Bandwidth		20MHz −3dB	25MHz −3dB		
Characters to Display		2,000 Characters (80×25 Lines)	4,050 Characters (90×45 Lines) or 2,000 Characters (80×25 Lines)		7,200 Characters (120×60 Lines) (Ex.)
Active Display Area		H. 220㎜ (Max. 240㎜), V. 137㎜ (Max. 150㎜)			
Active Display Time		H.45μsec., V. 13.1 msec.	H. 30.2μsec., V. 13.1 msec.		H.23.8μsec., V. 13.1 msec.
Resolution		640 Dots× 200Lines(Max. 280Lines)	640 Dots × 400Lines(Max. 450Lines)	640Dots(Max. 800Dots)× 400Lines(Max. 450Lines)	720Dots(Max. 800Dots)× 512Lines(Max. 550Lines)
Amplifier Type		Linear Amplifier			
Connector Type		9P D-SUB(1m)			
Power Supply		AC 120V (85~132V) AC 220V (180~260V) 60Hz (50~65Hz)			
Power Consumption		Approx. 70W	Approx. 80W		
Environment	Operating Temperature	0°C~+50°C			
	Storage Temperature	−10°C~+65°C			
	Storage Humidity	20%~80% (Non-Dew)			
Weight		Approx. 10kg net			
External Size		314(W)×322(H)×354(D)㎜			
Size of Packing	Single carton box	443(W)×442(H)×403(D)㎜			
	Double carton box	518(W)×532(H)×478(D)㎜			
FCC		Class A approved (Class B Complied)			

特約經銷商

萬達電腦公司

深水埗黃金商場地庫21號

電話：3-610095

* IBM PC、PC/XT is registered trademark of International Business Machine Corp.

** Apple II is a registered trademark of Apple Computer Inc.

VERY MONTH IS DIFFERENT!

COMPUSOFT 磁碟月刊

內容大革新
售價大革新

COMPUSOFT磁碟月刊，每期均出版兩張5¼"雙面雙密的磁碟(合共四面)，以DOS 3.3 FORMAT，適用於任何蘋果電腦或相容的機種。

COMPUSOFT磁碟月刊每期均包羅有趣豐富、刺激拿並的實用工具程式、電腦自學文章，最新軟件測試報告、適合任何行業的實用遊戲、教育軟件……等精彩內容，是一本適合電腦發燒友、未來電腦從業員、學校教職員，學生，家長及有志投身電腦軟件工業人士的必備電腦雜誌。

磁碟月刊每期所包含內容的技巧，更是透過一羣專業程序設計師精心製作的成果，它有極大的研究價值，是用家們值得珍藏之電腦雜誌。

馬備COMPUSOFT磁碟月刊，只花小小代價，獲多多的得益。

SOFT
碟月刊
)

地鐵／九廣鐵路

豪電視服務站特約發售

磁碟月刊第十二期（十一月廿日出版）

今期月刊出版的內容題材廣泛，技巧高超。其中包括：

每期動畫設計精選：今期劇目是「好奇之過」，大家都心欣賞我們的主角的神奇遭遇。

PRINTSHOP 圖案精選：每期我們都爲大家設計一集精緻的圖案供 PRINTSHOP 使用，今期主題是有關音樂的圖案。

■實用程式：

(1)電子作曲器：這個作曲工具簡單易用，效果比美 MUSIC CONSTRUCTION SET 及其他的專業作曲軟件，是一個不可多得的工具。

(2)高解像圖編輯工具：利用本工具，可以爲任何的高解像圖進行修改編輯，較 BEAGLE BR-OTHER 的 APPLE MECHANIC 容易用，而且控制穩定，能進行細緻圖案修補。

(3)畫圓程式：利用多種編寫程式技巧，提供多種表達方式在高解像頁裡畫出大小不同的圖形，既實用又有參考價值。

(4)目錄選擇執行程式：本工具程式是用來作爲一個用左右箭咀選擇的 HELLO 程式，它的優點是同類型程式之冠，例如可自由轉換 DRIV-E，顯示 FREE SECTOR 數值，再讀C-ATALOG 及在程式 FILENAME 旁指示出該 FILE 是屬何種類型檔案，此電腦時代刊出過的更好。

■教育程式：

(1)數學幾何公式集：這個公式集收集了廿三種計算各種不同幾何圖形問題，是同學的好幫手。

(2)迴數平方計算示範：迴數是數學科中一種非常堪玩味的問題，所謂迴數是指一個數無論是由左邊讀起，或由右邊讀起都是一樣的數，例如123321，便是一個迴數。

(3)超級拼字訓練：專供小孩學習英文單字的程式，家長們可按兒童的程度自己設計輸入生字。

(4)作故事：電腦預先編撰一些故事大綱，然後由玩者自己照電腦指示隨意輸入各名詞／形容詞，最後，電腦會按你的輸入作出一篇令人捧腹大笑的文章。

心理測驗：今期的主題是考驗閣下是否擁有超能預感能力，測驗分兩部進行。大家可試找出在這方面的能力到底有多強。

■精選遊戲：

(1)月球行車：一個高難度的遊戲，玩者要在月蝕之前進行搶救同伴任務，玩法非常新穎。

(2)捉虫競賽：仿效金石計數機而比它難度高萬倍的遊戲，遊戲有三個 LEVEL，我們相信你會很難一次過玩完三關。

(3)波子機：另一個不同圖形設計的波子機，十分耐玩。

此外尚有其他內容，未能一一盡錄。

內容……大革新
訂價……大革新

COMPUSOFT
磁碟月刊

由第十二期開始
以全新姿態出現

COMPUSOFT 磁碟月刊自出版以來，經已有一年歷史。在過去一年內，我們不斷努力改進，務求令磁碟月刊的內容更加豐富，更加充實。事實上，從讀者的數目不斷增加，就足証我們的心血並沒有白費。

爲慶祝 COMPUSOFT 磁碟月刊出版一週年，我們決定來一個革命性的突破：我們決定從第十二期開始，將 COMPUSOFT 磁碟月刊的售價大幅的減低，由每期零售價 50 元一套減爲 35 元，長期訂閱費亦由一年十二期的 180 元減爲 120 元。同時爲保持質量，我們依舊選用更高質素的磁碟，而每期依然保持出版兩張雙面灌錄內容的磁碟。

我們此舉做法，目的是令 COMPUSOFT 磁碟月刊更普及更迎合讀者的消費能力。讀者們只要付出非常少的代價，就有多多的得益。現在就請閣下立即辦理訂閱或到各地鐵電視服務站選購。

訂閱辦法

訂閱磁碟月刊辦法簡單，只需填妥附同的訂閱表格，連同劃線支票或郵政滙票（滙票請勿劃線）寄來九龍中央郵政信箱 71193 號，電腦時代出版社收，支票收款人請填上「電腦時代出版社」或「COMPUTING AGE PUBLISHER」。信封面請註明「訂閱磁碟月刊」。

香港本土以外的讀者只接受連同磁碟的訂閱，請附加空郵費 30% ，平郵加附 10% 郵費。

新訂戶如選擇不連磁碟的訂閱方法，請連同兩張雙面有效的空白磁碟（請貼上附有姓名及電話的LABEL），跟訂閱表格及訂閱費一併寄來，或多付 20 元購買兩張磁碟作爲開首訂閱之第一期使用。訂戶可選擇親身到服務部取碟或由我們郵寄方式。

訂閱表格

本人訂閱 COMPUSOFT 磁碟月刊 由第____期開始

□十二期　$360（包括每期兩張磁碟，即共 24 張）
□十二期　$120（訂戶自己出磁碟）
□試閱一期 $ 35（包括兩張磁碟）

姓名：_____ 電話：_____

地址：_____

支票號碼# _____ 銀行# _____

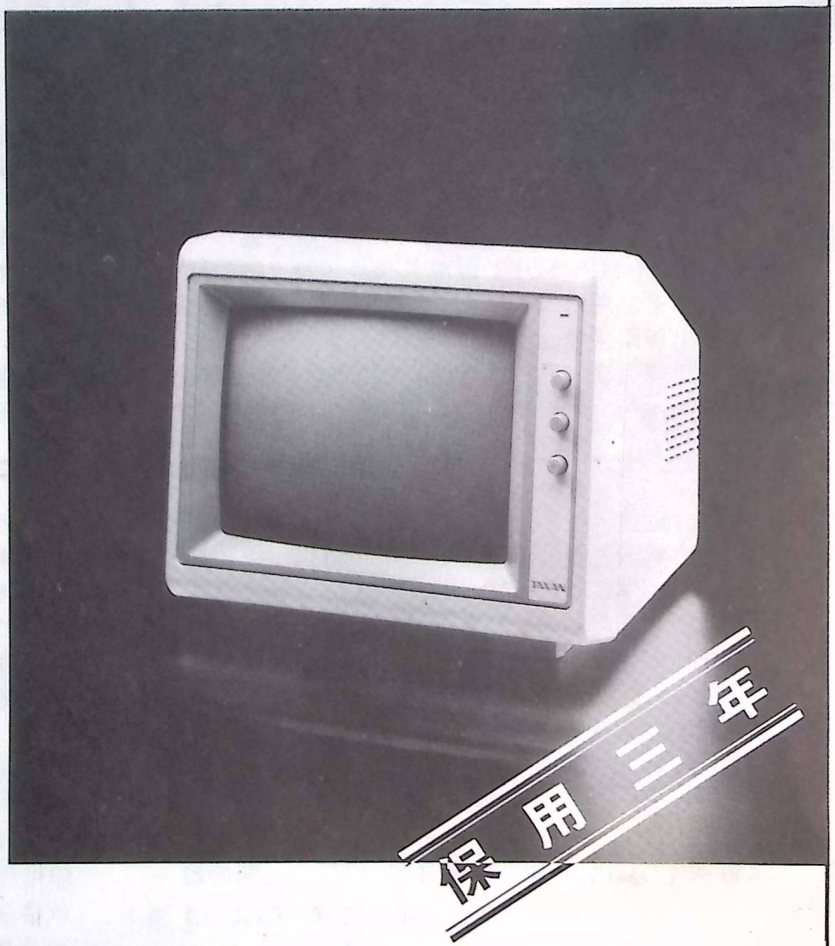
凡於一九八六年一月卅一日前訂閱 COMPUSOFT 磁碟月刊十二期，即可免費獲得 COMPUHOUSE 軟件屋最新出版的軟件：蘋果機械語言輸入編輯器，價值35元。

即時訂閱更可免費獲得價值35元的贈品

TAXAN®

「泰山」12吋綠色/琥珀色(橙色)電腦螢幕

KX-1212/1213



- 不反光、方型、平面、深灰底色（P39）長殘影顯像管。
- 視頻寬達20MHz以上。
- 解像力特高，畫面更清晰、穩定。
- 顯示字數2000字（80×25行）。
- 適合IBM PC* 、PC/XT* 及同類型產品。
- 全日本製造。

總代理：

VINCENT COMPUTER CENTRE LTD.

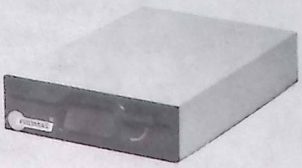
雲生電腦有限公司（請認明有限公司，並無分號）

九龍尖沙咀柯士甸路22-26號好兆年行608室 電話：3-7211288-0 電訊：HX38316 VCC HK

* IBM PC、PC/XT are registered trademark of International Business Machine Corp.

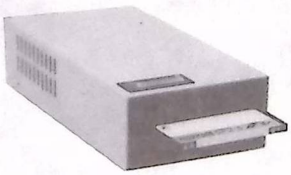
PROMAS 電腦產品

5.25"軟磁碟
雙面雙密度，適合IBM*、APPLE*II 等電腦使用。備有彩色碟或黑色碟，可供選擇。



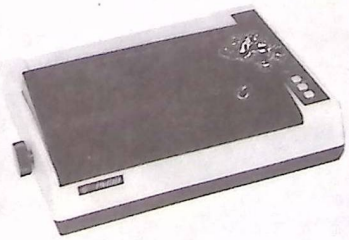
JT5.25"薄身磁碟機
多種「WRITE PROTECT」及「HEAD SPEED」選擇，適用於APPLE*II 及相融電腦。

JN磁碟機
傳統設計，適用於APPLE*II 及相容電腦。

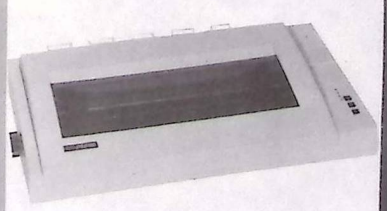


305 SE 3"磁碟機
多種保護資料功能，可與一般APPLE*II 磁碟機共用或單獨使用。

POWER ESCORT後備電源供應器
提供高質素電源；防止一般問題電源對電腦及精密儀器造成之破壞。多種輸出功率可供選擇。



PX－100打印機（9.5吋）
印字速度每秒100字，CENTRONIC 平行界面接口，適合一般電腦使用。



PX－136打印機（15吋）
印字速度每秒130字，CENTRONIC 平行界面接口，適合IBM*及相容電腦使用。

PF－150手提傳真機
携帶方便，可與世界各地傳真機互傳訊息。

總代理

PROMAS COMPUTER CENTRE LTD.

寶馬士電腦有限公司

九龍尖沙咀柯士甸路22－26號好兆年行608室　電話：3-674687　電訊：HX38316

* Registered trademark of Apple Computer Inc.

* Registered trademark of International Business Machine Corp.

OPERATIONAL PERFORMANCE

MODEL	Sides	Density	Sector	Track	TPI	Format	Capacity
MD/2D	2s	Double	Soft	35 or 40	48	None	500KBytes

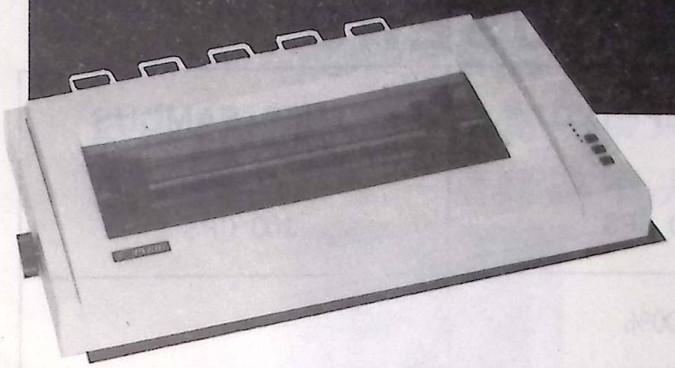
總代理：

PROMAS COMPUTER CENTRE LTD.

寶馬士電腦有限公司

九龍尖沙咀柯士甸路22－26號好兆年行608室　電話：3-674687　電訊：HX38316

選購打印機之前
必須考慮之要點：



1. 要選擇9吋或15吋打印機？

通常用戶只想購買一部9吋打印機來印一般來往信件，但當公司電腦化後，要編印一份詳細報告表或比較複雜的文件時，9吋打印機就不能勝任，到時便要更換一部15吋打印機。

PROMAS PX-136 15吋打印機每行可印68-233個大小文字。

2. 打印機印字速度要多快才適合？

印字速度當然越快越好，但是打印機價格也因速度而相應提高，一般用途的打印機速度每秒100字也足夠。

PROMAS PX-136印字速度每秒130字。

3. 打印機的墨帶壽命對印字成本有什麼影响？

墨帶壽命是直接影響印字成本。購買打印機只需付出一次費用，如果該打印機只能用一些壽命短的墨帶，日後所需的費用便非常龐大。

PROMAS PX-136使用特有新設計的「雙軌循環式墨帶」，可使墨帶壽命加長一倍，合乎經濟原則。

4. 打印機能否與一般電腦相容？

一般打印機只有普通的印字功能，不能印出一些在IBM* PC電腦中的特別圖案或半圖案字體，這也就是不能與該等電腦相容。

PROMAS PX-136內置獨有的軟件ROM，與IBM* PC完全相容，APPLEII PLUS及一般電腦也可使用。**

5. 要付出多少費用才可購買一部具備上述功能的打印機？

從前15吋打印機的價格均在港幣3000元以上，較9吋打印機昂貴，但得到的只是一般功能，而且墨帶價錢也非常昂貴。

PROMAS PX-136具備上述功能，售價與一般9吋打印機相差無幾。

如需要更詳細資料，請往本公司陳列室或各大零售商參觀操作示範。

總代理：

寶馬士電腦有限公司

PROMAS COMPUTER CENTRE LTD.

九龍尖沙咀柯士甸路22－26號好兆年行608室
電話：3-674687　電訊：HX 38316

*IBM PC is registered trademark of International Business Machine Corp.
**Apple II is a registered trademark of Apple Computer Inc.

Super 5

贈品

Super 5 EP-1201 精美防塵套
隨機附送

- 十月十五日起送完即止
- 盒面貼有Super 5 三週年紀念金色標貼

ATTN : IBM+ PC—XT USERS :

	SUPER 5 EP-1201 *	ANOTHER FAMOUS BRAND
SPEED	120 CPS	100 CPS
IBM COMPATIBILITY	100%	———
PRINT MODE SWITCH	YES	———
RIBBON LIFE	3 MILLION CHARACTER	1 MILLION CHARACTER
LETTER QUALITY MODE	PERFECT LETTER QUALITY (PLQ)	NEAR LETTER QUALITY (NLQ)
PROPORTIONAL MODE	YES	———
80/132 COLUMN SWITCHING	YES	———
IBM SPECIAL CHARACTERS & GRAPHICS SET	YES	———

BEFORE YOU BUY A PRINTER CHECK WHETHER IT IS PERFECT LETTER QUALITY & SUPPORTS FULL IBM—PC SPECIAL CHARACTER AND GRAPHIC SETS

*RATED BY U.S. CONSUMER REPORT JUNE 1985 ISSUE AS THE BEST OVERALL DOT MATRIX PRINTER

+IBM is a registered trademark of International Business Machines Corp.

A & N ENTERPRISE CO., LTD.
HONG KONG EI-EN SERVICE CENTRE
20/F., WESTERN CENTRE., 40-50 DES VOEUX ROAD, WEST, H.K.
TEL : 5-461179 5-488463 TELEX : 65218 MCLWW HX FAX : 5-497149

電腦功能人皆知
先租後買有保障

本港各行各業均趨向電腦化，但購置電腦系統除價錢昂貴外，更未必合用；再加上維修、員工訓練、軟件之配合等，每一項皆耗費大量金錢，故此很多公司對於電腦化計劃一直疑惑不決……

基於上述原因，本公司特推出電腦租用計劃，保証貴公司能以最低廉之價錢試用電腦，從而了解清楚電腦對貴公司之幫助及確信本公司服務完善，再決定是否購買如此便可避免因買錯電腦系統而招致金錢及時間之損失。
詳情如下：

1. 租　用　期：最短三個月，每個月只須付四佰餘元起，若三個月內決定購買，除可獲特價優待外，已付租金可自買價中減去。
2. 功　　用：文件處理、會計、CAD、生產及營業管理……等。
3. 保　　養：a) 租用期內即日修理或換機；
   b) 購買後可獲一年免費保養。
4. 其他服務：可爲貴公司訓練電腦操作員工，確保貴公司能利用電腦解決辦公室工作。

超卓電腦系統有限公司
SUPREMIA SYSTEMS LTD.

九龍深水埗福華街146號
黃金商場地庫20號室
電話：3-601531, 3-617206
3-612549

◀ 歡迎查詢或要求示範 ▶

香港教育司署立案（1957年創校）

世界函授學校

規模大　優點多　科目好

初級電腦科
電腦基本原理，蘋果牌，IBM PDP 等等電腦操作，程式編寫，八個月畢業。

初級無綫電科
無綫電基本原理，收音機製作及修理，全部圖解教導，十個月畢業。

半導體裝修科
講解最新半導體IC集成電路及TV身歷聲收音機擴音機原理，十個月畢業。

實用數理
可以加强數理基礎，講授幾何、代數、三角及計數機運用，六個月畢業。

$x+y=z$

實用商科
講解經濟學識、銀行學、進出口貿易、商業管理，學習易，六個月畢業。

無綫電工程
講解無綫電綫路設計發射機及天綫設計，是深造的理想科目，一年半畢業。

汽車工程
最新汽車構造、內燃機原理，自動波風油軑動力剎制等原理，一年半畢業。

冷氣工程
空氣調節原理，設計及檢修，冷藏及製冰原理，冷氣機檢修，一年半畢業。

商業美術科
商業美術基本理論和技術，廣告設計，黑白畫圖案畫技術，十二個月畢業。

應用漫畫
繪畫基本技術、人物動作及表情畫法，活動卡通，世界漫畫，六個月畢業。

學習時間
函授時間是沒有時間規限如時間充裕可提早完成所學的科目，提前畢業。

導師
國內外著名大學學士，經香港教育司考核合格，教導認眞，決不馬虎。

不收雜費
學生只需要交低廉之學費外，所有課本，試題及測驗紙均免費派發。

文憑
本校頒發文憑，各大機構認許，實是最佳學歷證件。

本港學生可利用電話詢問索取章程　電話③－803532

請依綫剪下，填好姓名，回郵地址，貼在信封上投寄，本校即免費寄上簡章

郵件請寄：P.O.BOX 70011, KOWLOON, CENTRAL POST OFFICE
九龍中央郵政局信箱 70011 號

技術就是力量

（請註明要學的科目）

- □初級電腦科
- □無綫電工程科
- □初級無綫電科
- □半導體裝修科
- □實用數理科
- □汽車工程
- □冷氣工程
- □實用商科
- □商業美術
- □應用漫畫

（寄自）

姓名：

地址：

請貼上郵票

香港教育司署立案

To：世界函授學校註册處

P. O. BOX No. 70011

KOWLOON CENTRAL POST OFFICE,

HONG KONG.

香港九龍中央郵局信箱70011號

電話：K-803532

學問就是財富

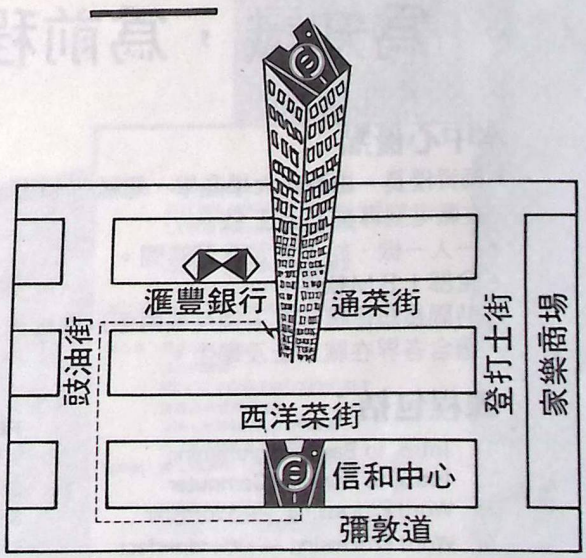
何必自己動手

免費 由專業人員上門爲你修理微型電腦

請即加入本公司之保養及免費維修服務

24小時熱線電話服務

3-848129



微型電腦保養維修中心

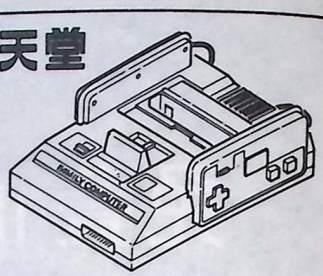
ROM & RAM computer ltd

本公司現已搬往旺角通菜街1A -1J 威達
商業大厦19/F , 1905室 TEL：3-848129

OFFICE: Rm. 1905 Witty Coml. Bldg., 1A—1J, Tung Choi St., Kowloon. TEL : 3-848129

Telex: 57551 ROMHX

任天堂



任天堂遊戲機 現時全日本最流行及最勁之遊戲機，大部份遊戲都是街機遊戲。機及遊戲卡均售。

MSX Computer 及遊戲卡, ROM Card, RAM Card.

Mocking Card $220

使 AP II 所產生之音响效果達到現場一樣，有眞實感 ，尤其配合 Sky Fox 及 Music Construction Set 與街機毫不遜色。現時可配合用之 Software 已經非常之多，將來且有更多的軟件支持，其它非原裝線路之同類卡，有部份軟件不能操作及效果差的現象經常發生。

Speech Chip $ 250 連碟連說明書，只須一片已可使用。

Track Reader $ 200

磁碟讀軌器，能讀出 ½Track，可看到磁頭走動之情況，用以配合抄碟，更可達到最佳效果及可以減少抄碟的時間。

精美電子公司 JAN MI ELECTRONIC CO. 九龍深水埗福華街黃金商場地庫48號A

Shop 48A, Basement, 146-152 Fuk Wa Street, Golden Center, Golden Building Kowloon.

博悅電腦訓練中心

爲知識，爲前程 來博悅，學電腦

本中心優點：

- 師資優良，由英美大學商學，電腦等碩士、學士及在職電腦專業人士任教。
- 一人一機，給予充足實習時間。
- 全部 I B M軟件。
- 時間長短任擇，有一個月或兩個月課程。
- 適合各界在職人士及學生。

課程包括：

		FEE
1)	Intro. to Basic Programming	$350
2)	Intro. to Business Computer	$500
3)	Word Processing — Wordstar	$390
4)	Word Processing — Wordperfect	$390
5)	Database — dBasell & III	$490
6)	商業應用中文電腦	$600
7)	Spreadsheet — Lotus 1–2 –3	$495
8)	Spreadsheet — Symphony	$800
9)	Intro. to Cobol	$450
10)	Advanced Cobol	$450
11)	Intro. to Pascal	$395

地址：灣仔軒尼詩道458號，金聯商業大厦801室 電話：5-8910119 5-8910315

（近地下鐵銅鑼灣站，波斯富街出口）

一輯電腦用家
不容錯過之必讀物

GPLE 使用方法深入研究

1. GPLE 的概念
2. 載入及執行 GPLE
3. 編輯模式
4. GLOBAL EDITING (全域性程式編輯模式)
5. LISTING 的控制
6. 大小楷模式輸入
7. 80行咭的相融
8. 鍵盤特置的緩衝設施
9. ESCAPE 功能之設置
10. GPLE的附隨修正程式
11. 維護GPLE
12. 記憶體配置圖

附操控指令總表

特約經銷處：
- 電腦時代讀者服務部
- 中環三聯書局
- 創業雜誌中心 西環屈地街22號地下（明愛中心對面）
- 深水埗黃金商場地庫 21
- 資訊電腦中心 北角英皇道121號七海商業中心134室

電腦技術應用叢書經已面世

128K RAM咭 應用專輯

第一章 序言
第二章 選購篇
- 2.1 前言
- 2.2 買成品之選擇
- 2.3 自嵌者之選料及裝嵌
- 2.4 零件表
- 2.5 規格說明
- 2.6 總結

第三章 測試及使用篇
- 3.1 前言
- 3.2 在DOS 3.3測試
- 3.3 在CP/M之測試
- 3.4 在PASCAL 之測試

第四章 硬體改良篇
- 4.1 前言
- 4.2 原理概述
- 4.3 改良步驟
- 4.4 總結

第五章 INSPECTOR與128K RAM CARD
- 5.1 系統軟件之使用
- 5.2 FAST DOS 與 128K RAM CARD
- 5.3 PRINTER BUFFER
- 5.4 總結

第六章 硬件篇
- 6.1 零件概述 (4164、3242、74LS 93)
- 6.2 工作原理

MOCKINGBOARD 技術應用

第一章 MOCKINGBOARD和聲音關係
- INIT 開啓 MOCKINGBOARD 程式
- LATCH 設定記存器工作副程式
- WRITE 輸入資料副程式
- RESET 清除記存器數值副程式
- TABLE ACCESS 聲音表副程式

第二章 MOCKINGBOARD 基本運用
- ENVELOPE PERIOD 和 ENVELOPE SHAPE

第三章 聲音效果的測試
- 聲音的混合
- MUSIC CONSTRUCTION SET
- 直接影响音樂的變數
- 鳥鳴，槍聲和背景音樂的混合
- 海浪聲和背景音樂的混合

第四章 擴大MUSIC CONSTRUCTION SET
- 一隻磁碟可以存放幾多首樂曲？
- 反覆依次演奏問題
- MUSIC LIST CREATOR
- 演奏更長的樂曲

第五章 把聲音和音樂加入程式實例
- (一)高空擲物 DANGER ROAD
- (二)空戰靈魂 AIRFIGHT SHOOT OUT
- (三)太空侵略者封閉戰 CLOSING ALIEN

一九八五年七月底出版

電腦時代出版社

最緊要好學

一人一部
其樂無窮

弘智電腦
直接學習

弘智電腦智識中心
香港軒尼詩道467號
建德豐商業大廈703室
銅鑼灣地鐵站隔鄰

5-752803

•特設團體租用時間歡迎垂詢.

English Vocabulary Basic Data Base II English Vocabulary Basic Data Base II English Vocab
Tutor Bagels Pascal Multiplan Typing Tutor Bagels Pascal Multiplan Typing Tutor
Spelling Bee CP/M Word Star Spelling Bee CP/M Word Star Spelling Bee CP/M
Basic Introductory Course Mathematics Multiplan Basic Introductory
Advanced Basic Apple Work Logo Advanced Basic Apple Wo
Advanced Pascal Word Star Bagels Pascal Advanced Pascal
Data Base IL English Vocabulary Spelling Bee Logo
Multiplan Basic Introductory Course Typing Tutor
Star Advanced Basic Data Base II Word St
Mathematics Pascal Apple Work
Basic Data Base II English Vocal
Multiplan Typing Tutor
Spelling Bee CP/M
Introductory
Apple Wo
Pascal
Logo

發揮dBASE II最大優點的

發票資料檔案系統

弘智電腦智識中心供稿／作者：陳毅艮

微型電腦顧名思義具有體積細小，操作簡易等特色。但也同時兼具了貯存資料有限的弱點。要使微型電腦更好地爲用家效勞，如何掌握好有效的操作系統即能夠令電腦快速操作，同時又不會因程式複雜，致佔用貯存空間太多，實在是電腦程式設計人員不可迴避的重要課題。

在芸芸電腦程式中，應用於微型電腦的可謂數不勝數。本中心的主要着眼點在於實際應用，故對於程式的基本要求是易學易用，快速有效。總之，絕不提倡曲高和寡的高調程式。

適用於蘋果電腦的資料庫（dBASE II）系統正具備了這一切條件。因此，本中心樂意推介運用資料庫手法編寫的一些實用的電腦程式，供用家使用。

正如上文開宗明義點出了微型電腦貯存空間有限的特性，各讀友參考下文介紹之程式時，請留意我們刻意地要節省空間的主要重點。

資料庫的設計

一個良好的資料庫設計是以方便程序的編寫，資料的存取及搜尋，良好的執行效率及速度，還要顧及貯存空間的運用等等爲主要目的。以下是以一間公司的發票系統爲例，解釋一下怎樣去設計一個資料庫。

INVOICE # :_____ DATE :_____

SOLD BY :_____

ITEM NO.	DESCRIPTION	QUANTITY	PRICE	TOTAL

一般發票樣品

第一個步驟是要了解清楚這個系統在人手操作條件下是怎樣運行的。

第二個步驟是要將發票轉成資料庫檔案。

假設每一張發票可以有五欄空間來填寫銷貨資料。因此每一筆記錄將會用去24＋48×５＝264位元貯存空間。如果平均每一張發票只用去兩欄銷貨資料空間，每一筆記錄便會消耗了48×３＝144位元組貯存空間。這是一個極大的浪費；除了浪費空間外，還會降低了資料的存取，搜尋的速度。因爲一個資料庫的大小會直接影响資料庫程序的執行速度。

發票資料檔案

列表㈠是一個發票資料檔案，是利用dBASE II寫成。大家可發覺到列表㈠的資料庫設計佔據貯存空間過大，因此我們將發票資料庫分成兩個檔案，分別是發票（列表二）及貨物銷貨檔（列表三），且看是否可以節省一點。

根據這個新的設計，平均以一張發票平均佔用兩個銷貨檔案記錄，共用去27＋２×55＝137位元貯存空間，比以前一種設計節省了264－137＝127 位元空間，對於一間每天開一千張發票的公司來說，每天可節省一百多K的貯字空間。這可是個增加經濟效益的設計改進。

現在用兩個不同的檔案來儲存發票，看起來可能會增加了存取的操作程序。但是事實上，情況並不一定如想像中複雜，由於dBASE II 能夠爲我們提供強大的功能和指令，而且這兩個檔案中，那些記錄是屬於某一張發票。例如我們需要找尋一張發票，只要在主區域開啓發票檔，找到目標發票鍵入DISPLAY，然後在次要區域開啓銷貨檔鍵入DISPL－AY ALL FOR INVOICE＝P・INVOICE 便可得到整張發票。

要更進一步節省空間，我們可在銷貨檔案（列表三）中省去PRICE，或TOTAL 其中一欄，由於公式PRICE × QUANTITY＝TOTAL，可以計算出PRICE或TOTAL這一欄。因此平均每一張發票只佔用27＋２×45＝117 位元貯存空間。

更進一步的發展

每一間公司，一定有一個存貨部門。我們可以增加以下一個資料檔案（即存貨檔，見列表四）。

有了列表四這個存貨檔案，我們可以在開立發票之前先檢查公司的存貨是否足夠。以及知道那一種貨物需要補充訂貨。除此之外，我們還可以進一步縮短存檔中相同的ITEM：NO找到DESC 這一欄的資料。因此，平均每一張發票只佔用27＋２×25＝77位元組空間。

以下是介紹怎樣編寫程序來連結這幾個資料庫。

化零爲整的主程式

這個程式共有三個主要檔案用來紀錄資料，分別是存貨資料檔（列表五），銷貨資料檔（ITEM・DBF）（列表六）及發票檔（INVOICE・DBF）（列表七）。結構如下：

除此之外，還有三個暫存檔案，第一個是列印檔（PRI－NT・DBF）（列表八）用來印INVOICE，結構如下見列表八。

第二個是報告檔案（列表九），用來印某一天的營業報告，結構見列表九。

第三個是暫存檔案（TRAWS・DBF）（列表十），用來紀錄發票上的銷貨資料。

最後我們還有一個報告檔案（DAILY・FRM）結構見列表十一。

輸入方法是鍵入MODIFY COMMAND DAILY・FRM；再依照以上的結構輸入。

在開始本程式之前，先要根據上面的檔案結構CREA－TE 6 個資料檔案，再引索幾個原始檔案，鍵入：

```
USE MST:ITEM
INDEX ON DESC TO MST:ITEM
USE ITEM
INDEX ON ITEM:NO TO ITEM
USE INVOICE
INDEX ON DATE TO INVOICE
```

在進入dBASE之前，一定要把眞實的日期輸入，因爲在發票上的日期是用這個日期的。只要鍵入DO MENU ，便可以開始本程式，跟着便可以見到主菜單，與及：

```
PICK A NUMBER(1,2,3,4,5 OR
6)OR <Q>UIT::
```

按Q可以離開本程式回到dBASE ，或選擇1至6的功能。

系統功能介紹

(一)開立發票

在主菜單鍵入1便可以開始開立發票，接着便可以見主日期及發票號碼。

```
ENTER YOUR NAME
```

輸入售貨員姓名後便可會見到。

```
ENTER ITEM NUMBER
```

輸入貨品編號後便可以見到貨品名稱及價錢，還可以見到。

```
IS THIS THE ONE YOU WANT (Y/N)
```

按N可以重新輸入貨品編號，按Y後在螢幕最後一行見到存貨數量。輸入售貨數量後，可以見到：

```
ANY MORE ITEM TO SELL?(Y/N) <E>XIT
```

按N可以回到主菜單，不儲存這張發票。

按Y可以繼續輸入新的貨物編號，每一張發票可以售出十一項不同的貨物，如果不夠用，只可以開立另一張發票，按N便會結束這張發票，這時可以見到：

```
DO YOU WANT TO PRINT THIS INVOICE?
<Y/N>
```

按Y可以把發票輸出列印機，然後回到主菜單；按N可以直接回到主菜單。

(二)檢查及列印發票

檢查及列印發票，在主菜單鍵入二便開始這項功能，跟着可以見到：

```
ENTER INVOICE TO VIEW/PRINT
```

輸入發票號碼，不久便可以見到整張發票，與及

```
DO YOU WANT TO PRINT ON PRINTER.
(Y/N)
```

按Y可以印出這張發票，然後回到主菜單；按N直接回到主菜單。

(三)加添貨品入存庫檔案

將新貨物輸入存庫檔案，在主菜單輸入三後便可以見到新的貨品編號與及：

```
ENTER DESCRIPTION,PRICE AND ON-HAND
```

輸入貨品名稱，價格及在存貨數目後，便可以見到

```
ARE YOU SURE?(Y/N)
```

按N後可以更改輸入，按Y後便可以見到

```
DO YOU REALLY WANT TO SAVE THIS
ITEM?(Y/N)
```

按Y便可以將新的資料存到存貨檔案上：按N不會將新資料儲存。在輸入貨品名稱，儲存數量及價格後，如果見到：

```
ITEM EXIOT! DO YOU WANT TO
CONTINUE,PRESS<N> TO EXIT.
```

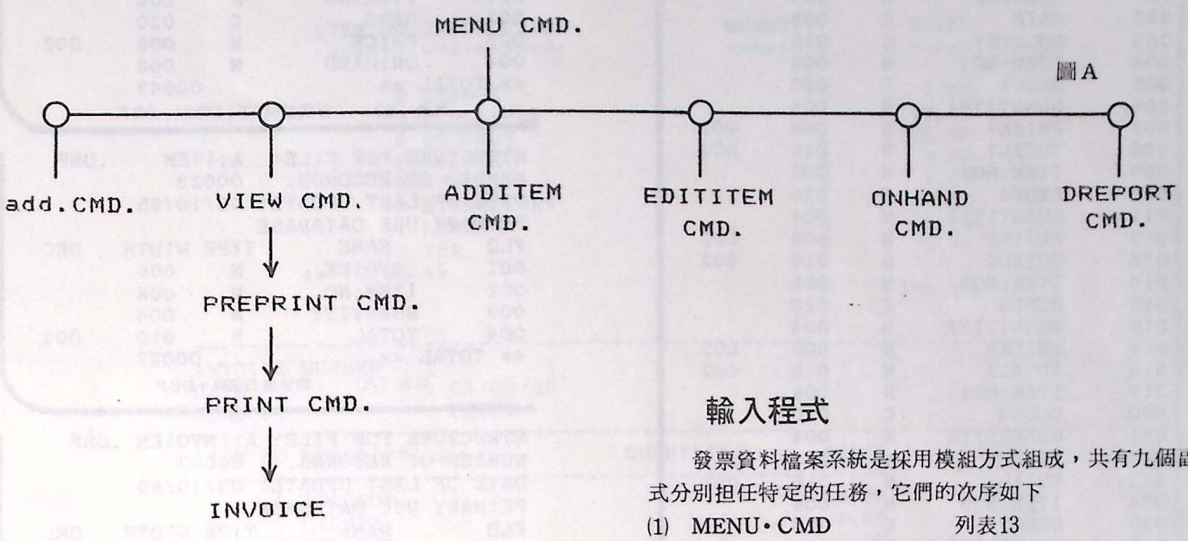
表示在存貨檔已經有這種貨品，應該按N回到主菜單。選擇功能五，增加存倉的數目，或按其他鍵，然後更改貨品的輸入。

(四)更改貨品售價

在主菜單中按四，便可以見到十六種貨品的名稱，編號，價錢及存貨數量。在螢幕最低一行可以見到：

```
<E>NTER ITEM NUMBER,e(X)it OR
<SPACE BAR SEE MORE.
```

發揮 dBASE II最大優點的發票資料檔案系統



圖A

按X可以回到程式主菜單。
按SPACE BAR 可以看到十六個以後的貨品名稱，售價，編號和存倉數量。
按E可以輸入貨品編號及新售價，如果輸入的貨品編號不存在便要重新輸入。如果沒有問題便可見到：

```
DO YOU WANT TO EDIT MORE?(Y/N)
```

按Y可以繼續或按N回到主菜單。

（五）增加存貨數量

在主菜單中按六便可以開始這項功能，接着可以見到：

```
ENTER THE DAY TO REPORT? YY/MM/DD
```

注意輸入日期的次序是年／月／日。不久之後便可以見到整篇報告表及

```
DO YOU WANT TO PRINT ON PRINTER?(Y/N).
```

按Y可使報告表印在紙上，然後回到主菜單。
按N可以直接回到主菜單。

基本上，本文所提出的發票資料檔案系統能適合一般公司使用，列表十二就是一個使用本系統編印出來的發票例子，值得讀友們參考。以下是本系統整個所包括的檔案總錄及使用方法。

輸入程式

發票資料檔案系統是採用模組方式組成，共有九個副程式分別担任特定的任務，它們的次序如下

(1) MENU・CMD 列表13
(2) ADD・CMD 列表14
(3) VIEW・CMD 列表15
(4) ADDITEM・CMD 列表16
(5) EDITITEM・CMD 列表17
(6) ONHAND・CMD 列表18
(7) DREPORT・CMD 列表19
(8) PREPRINT・CMD 列表20
(9) PRINT・CMD 列表21

圖A將以上9個程式的關係清楚列出，在功能上，MENU・CMD是主目錄及日期的控制程式，ADD・CMD是計算貨品數量和價格總數。

VIEW・CMD 是將發票資料庫內的單獨記錄列出，它會再呼叫PREPRINT・CMD及PRINT・CMD 兩個副程式來獲得發票上的資料及數據，並翻查舊記錄。

ADDITEM・CMD是將存貨種類增加之用，它負責將新資料添加到資料庫上。

EDITITEM・CMD 是負責修改貨品的價格，以適應市場需要。

ONHAND・CMD 是把存貨資料，價格及數量列出之用。

DREPORT・CMD 是把一個特定日期的所有銷貨記錄列出之用。

以上我已經把各個副程式的功能解釋過，各位若果自行鍵入的話，我建議最好依這個列表次序輸入，這樣在除錯時可以一個一個功能去做，不會一場胡塗！ ■

編者註：對於本文所介紹的發票資料檔案系統有興趣而又不想鍵入本文的讀友，我們同時出版本系統的磁碟版本，由於本系統是採用CP／M系統FORMAT ，故我們獨立出版一張程式磁碟（磁碟亦有兩種版本，分別爲：IBM PCD－OS FORMAT及APPLE CP／M FORMAT），每張售十五元。歡迎到讀者服務部選購，或利用郵購。訂有程式磁碟的長期訂戶，請寄來空磁碟一張，我們會免費COPY副本（只限APPLE FORMAT）。訂戶請連同放得下磁碟的回郵信封一個（貼一元郵票）寄來，或親身到讀者服務部辦理。 ■

發揮 dBASE II最大優點的發票資料檔案系統

```
STRUCTURE FOR FILE:   B:INVOICE .DBF
NUMBER OF RECORDS:    00000
DATE OF LAST UPDATE: 00/00/00
PRIMARY USE DATABASE
FLD       NAME       TYPE WIDTH    DEC
001      INVOICE       N    006
002      DATE          C    008
003      SOLD:BY       C    010
004      ITEM:NO1      N    006
005      DESC1         C    020
006      QUANTITY1     N    004
007      PRICE1        N    008    002
008      TOTAL1        N    010    002
009      ITEM:NO2      N    006
010      DESC2         C    020
011      QUANTITY2     N    004
012      PRICE2        N    008    002
013      TOTAL2        N    010    002
014      ITEM:NO3      N    006
015      DESC3         C    020
016      QUANTITY3     N    004
017      PRICE3        N    008    002
018      TOTAL3        N    010    002
019      ITEM:NO4      N    006
020      DESC4         C    020
021      QUANTITY4     N    004
022      PRICE4        N    008    002
023      TOTAL4        N    010    002
024      ITEM:NO5      N    006
025      DESC5         C    020
026      QUANTITY5     N    004
027      PRICE5        N    008    002
028      TOTAL5        N    010    002
** TOTAL **                00265
```

列表（一）：普通發票資料檔案系統

```
STRUCTURE FOR FILE: A:INVOICE .DBF
NUMBER OF RECORDS:    00007
DATE OF LAST UPDATE: 03/10/85
PRIMARY USE DATABASE
FLD       NAME       TYPE WIDTH    DEC
001      INVOICE       N    006
002      DATE          C    008
003      SOLD:BY       C    010
** TOTAL **                00025
```

列表（二）：發票檔

```
STRUCTURE FOR FILE:   A:ITEM     .DBF
NUMBER OF RECORDS:    00023
DATE OF LAST UPDATE: 03/10/85
PRIMARY USE DATABASE
FLD       NAME       TYPE WIDTH    DEC
001      INVOICE       N    006
002      ITEM:NO       N    006
003      DESC          C    020
004      QUANTITY      N    004
005      PRICE         N    008    002
006      TOTAL         N    010    002
** TOTAL **                00055
```

列表（三）：銷貨檔

```
STRUCTURE FOR FILE:   A:MST:ITEM.DBF
NUMBER OF RECORDS:    00014
DATE OF LAST UPDATE: 03/10/85
PRIMARY USE DATABASE
FLD       NAME       TYPE WIDTH    DEC
001      ITEM:NO       N    006
002      DESC          C    020
003      PRICE         N    008    002
004      ON:HAND       N    008
** TOTAL **                00043
```

列表（四）：存貨檔

```
STRUCTURE FOR FILE:   A:MST:ITEM.DBF
NUMBER OF RECORDS:    00014
DATE OF LAST UPDATE: 03/10/85
PRIMARY USE DATABASE
FLD       NAME       TYPE WIDTH    DEC
001      ITEM:NO       N    006
002      DESC          C    020
003      PRICE         N    008    002
004      ON:HAND       N    008
** TOTAL **                00043
```

列表（五）：存貨檔MST：ITEM・DBF

```
STRUCTURE FOR FILE:   A:ITEM     .DBF
NUMBER OF RECORDS:    00023
DATE OF LAST UPDATE: 03/10/85
PRIMARY USE DATABASE
FLD       NAME       TYPE WIDTH    DEC
001      INVOICE       N    006
002      ITEM:NO       N    006
003      QUANTITY      N    004
004      TOTAL         N    010    002
** TOTAL **                00027
```

列表（六）：銷貨檔ITEM・DBF

```
STRUCTURE FOR FILE: A:INVOICE .DBF
NUMBER OF RECORDS:    00007
DATE OF LAST UPDATE: 03/10/85
PRIMARY USE DATABASE
FLD       NAME       TYPE WIDTH    DEC
001      INVOICE       N    006
002      DATE          C    008
003      SOLD:BY       C    010
** TOTAL **                00025
```

列表（七）：發票檔INVOICE・DBF

```
STRUCTURE FOR FILE:   A:PRINT    .DBF
NUMBER OF RECORDS:    00003
DATE OF LAST UPDATE: 03/10/85
PRIMARY USE DATABASE
FLD       NAME       TYPE WIDTH    DEC
001      ITEM:NO       N    006
002      DESC          C    020
003      QUANTITY      N    010
004      PRICE         N    010    002
005      TOTAL         N    010    002
006      INVOICE       N    006
** TOTAL **                00063
```

列表（八）

```
STRUCTURE FOR FILE:   A:DAILY    .DBF
NUMBER OF RECORDS:    00008
DATE OF LAST UPDATE: 03/10/85
PRIMARY USE DATABASE
FLD       NAME       TYPE WIDTH    DEC
001      ITEM:NO       N    006
002      DESC          C    020
003      QUANTITY      N    008
004      TOTAL         N    010    002
** TOTAL **                00045
```

列表（九）

```
STRUCTURE FOR FILE:   A:TRANS    .DBF
NUMBER OF RECORDS:    00003
DATE OF LAST UPDATE: 03/10/85
PRIMARY USE DATABASE
FLD       NAME       TYPE WIDTH    DEC
001      INVOICE       N    006
002      ITEM:NO       N    006
003      QUANTITY      N    004
004      TOTAL         N    010    002
** TOTAL **                00027
```

列表（十）

發揮 dBASE II最大優點的發票資料檔案系統

PAGE NO. 00001
01/09/85

DAILY REPORT

ITEM NO	DESCRIPTION	QUANTITY	TOTAL
2	ORANGE	2	2.80
3	PINEAPPLE	1	15.00
6	COCONUT	1	55.00
7	LYCHEE	2	3.60
8	APRICOT	2	4.00
10	STRAWBERRY	11	77.00
11	LEMON	4	6.00
12	TEA	3	11.70
** TOTAL **			175.10

INVOICE NUMBER : 1
DATE : 01/09/85
SOLD BY : PETER

ITEM NO	DESCRIPTION	QUANTITY	PRICE	TOTAL
6	COCONUT	1	55.00	55.00
7	LYCHEE	2	1.80	3.60
8	APRICOT	2	2.00	4.00
10	STRAWBERRY	1	7.00	7.00
		TOTAL :		69.60

列表（十二）：利用本系統所印出來的發票模樣。

```
1:
2: Y
3: <                    DAILY REPORT
4: N
5: Y
6: N
7: 8,ITEM:NO
8: ITEM NO;=======
9: N
10: 20,DESC
11: <DESCRIPTION;============
12: 10,QUANTITY
13: >QUANTITY;========          列表（十一）
14: N
15: 10,TOTAL
16: >TOTAL;======
17: Y
```

```
Listing of: B:MENU.CMD
 1  * menu.cmd
 2  SET ESCA OFF
 3  SET TALK OFF
 4  USE INVOICE
 5  GOTO BOTTOM
 6  STORE F TO DONE
 7  STORE $(DATE(),7,2)+"/"+$(DATE(),4,3)+$(DATE(),1,2) TO IDATE
 8  STORE DATE TO MDATE
 9  ERASE
10  USE
11  DO WHILE .NOT. DONE
12    IF IDATE < MDATE
13      @ 12,10 SAY "The date must later than YY/MM/DD "+MDATE
14      @ 10,10 SAY "Please enter today's date. YY/MM/DD ";
15            GET IDATE PICTURE '@@/@@/@@'
16      READ
17    ELSE
18      STORE T TO DONE
19      STORE $(IDATE,7,2)+"/"+$(IDATE,4,3)+$(IDATE,1,2) TO IDATE
20      SET DATE TO &IDATE
21    ENDIF
22  ENDDO
23  DO WHILE T          *endless loop for main menu until Press 0
24    ERASE
25    @ 1,20 SAY 'M A I N    M E N U'
26    @ 1,60 SAY 'Today is ' + DATE()
27    @ 2, 0 SAY '----------------------------------------';
28              + '----------------------------------------'
29    @ 4,10 SAY '(1) ADD INVOICE'
30    @ 6,10 SAY '(2) VIEW/PRINT INVOICE'
31    @ 8,10 SAY '(3) ADD ITEMS'
32    @ 10,10 SAY '(4) EDIT ITEMS PRICE'
33    @ 12,10 SAY '(5) ADD ITEMS ON HAND'
34    @ 14,10 SAY '(6) DAILY REPORT'
35    @ 16,10 SAY '(0) QUIT'
36    @ 18, 0 SAY '----------------------------------------';
37              + '----------------------------------------'
38    @ 20,10 SAY 'Pick a number (1,2,3,4,5 or 6)  or (0)uit'
39    STORE ' ' TO ANS
40    DO WHILE .NOT. ANS $ '1234560'       * limit the input as 1234560
41      STORE ' ' TO ANS
42      @ 20,52 GET ANS PICTURE '!'
43      READ
44    ENDDO
45    DO CASE                * choose the right sub-routine
46    CASE ANS = '1'
47      DO ADD
48    CASE ANS = '2'
49      DO VIEW
50    CASE ANS = '3'
51      DO ADDITEM
52    CASE ANS = '4'
53      DO EDITITEM
54    CASE ANS = '5'
55      DO ONHAND
56    CASE ANS = '6'
57      DO DREPORT
58    CASE ANS = '0'
59      RETURN
60    ENDCASE
61  ENDDO       * do while t
```

列表 13

列表 14

```
Listing of: B:ADD.CMD
 1  * add.cmd
 2  USE INVOICE
 3  GOTO BOTTOM     * calculate invoice number
 4  STORE # + 1 TO MINVOICE
 5  USE TRANS
 6  DELETE ALL
 7  PACK
 8  SELECT SECONDARY    * open the mst:item in secondary area
 9  USE MST:ITEM
10  STORE '          ' TO MSOLD:BY
11  ERASE
12  @  0,25 SAY 'A D D     I N V O I C E'
13  @  1, 0 SAY '------------------------------------';
14            + '------------------------------------'
15  @  5, 0 SAY '------------------------------------';
16            + '------------------------------------'
17  @  2, 0 SAY 'INVOICE      : ' + STR(MINVOICE,8,0)
18  @  3, 0 SAY 'DATE         : ' + DATE()
19  @ 22,10 SAY 'Enter your name.'
20  @  4, 0 SAY 'SOLD BY      ' GET MSOLD:BY
21  READ
22  @ 22,10
23  @  6, 0 SAY 'ITEM NO       DESCRIPTION              QUANTITY       PRICE';
24            + '     TOTAL'
25  @  7, 0 SAY '------------------------------------';
26            + '------------------------------------'
27  @ 20, 0 SAY '------------------------------------';
28            + '------------------------------------'
29  STORE 7 TO ROW
30  STORE F TO DONE
31  STORE 0 TO SUM
32  DO WHILE (.NOT. DONE) .AND. (ROW<19)    * repeat until end of this invoice
33    STORE 0 TO MITEM:NO, MQUANTITY, MTOTAL
34    STORE ROW + 1 TO ROW
35    STORE F TO FOUND
36    SELECT SECONDARY
37    DO WHILE .NOT. FOUND                  * check valid item number
38      @ 22,10 SAY 'Enter intem number.'
39      @ ROW, 0 GET MITEM:NO PICTURE '@@@@@@'
40      READ
41      CLEAR GET
42      LOCATE FOR ITEM:NO = MITEM:NO
43      STORE ' ' TO ANS
44      IF EOF
45        @ 22, 0
46        @ 22,10 SAY 'Item not exist, please re-enter!'
47      ELSE          * display description and price
48        @ ROW,14 SAY DESC
49        @ ROW,50 SAY PRICE
50        @ 22, 0
51        @ 22,10 SAY 'Is this the one you want? (Y/N) ' GET ANS PICTURE '!'
52        READ
53        CLEAR GET
54        IF ANS = 'Y'
55          STORE T TO FOUND
56          @ 22, 0
57          @ 22, 10 SAY 'Only ' + STR(ON:HAND,6,0) + ' on hand. Enter quantity. '
58        ENDIF
59      ENDIF
60    ENDDO
61    STORE F TO OK
62    DO WHILE .NOT. OK          * check enough item on hand, valid quantity input
63      @ ROW,36 GET MQUANTITY
64      READ
65      CLEAR GET
66      IF (MQUANTITY > ON:HAND) .OR. (MQUANTITY < 0)
67        STORE ' ' TO MESS
68        IF MQUANTITY > ON:HAND
69          STORE 'Not enough item, only '+STR(ON:HAND,6,0)+' on hand! ' TO MESS
70        ENDIF
71        IF MQUANTITY < 0
72          STORE 'Invalid quantity' ' + MESS TO MESS
73        ENDIF
74        @ 22, 0
75        @ 22, 0 SAY MESS + 'Please re-enter!'
76      ELSE
77        REPLACE ON:HAND WITH ON:HAND - MQUANTITY
78        STORE T TO OK
79      ENDIF
80    ENDDO
81    STORE PRICE * MQUANTITY TO MTOTAL
82    @ ROW,60 SAY MTOTAL
83    SELECT PRIMARY
84    APPEND BLANK                          * add item in transition file
85    REPLACE INVOICE WITH MINVOICE, QUANTITY WITH MQUANTITY, TOTAL WITH MTOTAL
86    REPLACE ITEM:NO WITH MITEM:NO
87    STORE SUM + MTOTAL TO SUM
88    @ 21,50 SAY 'TOTAL   : ' + STR(SUM,10,2)
89    STORE 0 TO TOTAL
90    STORE ' ' TO ANS
91    @ 22, 0
92    @ 22,10 SAY 'Any more item to sell? (Y/N) (E)xit ' GET ANS PICTURE '!'
93    READ
94    CLEAR GET
95    @ 22,10
96    IF ANS = 'N'
97      STORE T TO DONE
98    ENDIF                       * return the number of mst:item on hand
99    IF ANS = 'E'
100     GOTO TOP
101     DO WHILE .NOT. EOF
102       SELECT SECONDARY
103       LOCATE FOR ITEM:NO = P.ITEM:NO
104       REPLACE ON:HAND WITH ON:HAND + P.QUANTITY
105       SELECT PRIMARY
106       SKIP
107     ENDDO
108     CLEAR
109     RETURN
110   ENDIF
111 ENDDO                        * message say no free space in this invoice
112 IF ROW = 19
113   STORE ' ' TO ANS
114   @ 22, 0
115   @ 22, 0 SAY 'Please enter the items with a new invoice! ';
116             + 'Press any key to continue. ' GET ANS
117   READ
118 ENDIF
119 USE INVOICE INDEX INVOICE  * update the invoice file
120 APPEND BLANK
121 REPLACE INVOICE WITH MINVOICE, SOLD:BY WITH MSOLD:BY
122 REPLACE DATE WITH $(DATE(),7,2)+'/'+$(DATE(),4,2)+'/'+$(DATE(),1,2)
123 USE ITEM INDEX ITEM                  * update the item file
124 APPEND FROM TRANS FOR QUANTITY <> 0
125 STORE 'N' TO ANS
126 @ 22, 0
127 @ 22,10 SAY 'Do you want to print this invoice? (Y/N)' GET ANS PICTURE '!'
128 READ
129 IF ANS = 'Y'
130   USE
131   DO PREPRINT
132   SET EJECT OFF
133   SET PRINT ON
134   DO PRINT
135   SET PRINT OFF
136 ENDIF
137 CLEAR
```

發揮 dBASE II最大優點的發票資料檔案系統

列表 15

```
Listing of: B:VIEW.CMD
1   * view.cmd
2   SELECT SECONDARY
3   USE INVOICE
4   GOTO BOTTOM
5   STORE # TO MAX
6   STORE 0 TO MINVOICE
7   STORE F TO DONE
8   DO WHILE .NOT. DONE          * check if invoice exists
9     CLEAR GET
10    @ 20, 0
11    @ 20,10 SAY 'Enter invoice to view/print. ' GET MINVOICE
12    READ
13    IF (MINVOICE ) 0) .AND. (MINVOICE (= MAX)
14      STORE T TO DONE
15    ELSE
16      @ 20,.0
17      @ 20,10 SAY 'Invaild invoice. Please re-enter.'
18    ENDIF
19  ENDDO
20  DO PREPRINT
21  DO PRINT
22  STORE 'N' TO ANS
23  @ 22, 0
24  @ 22,10 SAY 'Do you want to print on printer. (Y/N) ' GET ANS PICTURE '!'
25  READ
26  IF ANS = 'Y'
27    SET EJECT OFF
28    SET PRINT ON
29    DO PRINT
30    SET PRINT OFF
31  ENDIF
32  CLEAR
```

列表 16

```
Listing of: B:ADDITEM.CMD
1   * additem.cmd
2   ERASE
3   @ 0,25 SAY 'A D D    I T E M S'
4   @ 2, 0 SAY '----------------------------------';
5            + '----------------------------------'
6   @ 12, 0 SAY '----------------------------------';
7            + '----------------------------------'
8   USE MST:ITEM
9   GOTO BOTTOM    * calculate new item number
10  STORE # + 1 TO MITEM:NO
11  SET INDEX TO MST:ITEM
12  STORE '                  ' TO MDESC
13  STORE 0 TO MPRICE, MON:HAND
14  STORE F TO DONE
15  DO WHILE .NOT. DONE                    * loop end until the input is ok
16    @  4,10 SAY 'ITEM NUMBER    : ' + STR(MITEM:NO,6,0)
17    @  6,10 SAY 'DESC           ' GET MDESC
18    @  8,10 SAY 'PRICE          ' GET MPRICE PICTURE '######.##'
19    @ 10,10 SAY 'ON HAND        ' GET MON:HAND PICTURE '########'
20    @ 14,10 SAY 'Enter description, price and on hand number.'
21    READ
22    FIND &MDESC
23    IF (# (> 0) .OR. (MPRICE ( 0) .OR. (MON:HAND ( 0)
24      STORE ' ' TO MESS
25      IF MPRICE ( 0
26        STORE 'Invalid price! ' TO MESS
27      ENDIF
28      IF MON:HAND ( 0
29        STORE 'Invalid on hand! ' + MESS TO MESS
30      ENDIF
31      IF (# (> 0)
32        STORE 'Item exist! ' + MESS TO MESS
33      ENDIF
34      STORE ' ' TO ANS
35      @ 14, 0
36      @ 14, 0 SAY MESS + 'Do you want to continue? Press (N) to exit ';
37              GET ANS PICTURE '!'
38      READ
39      @ 14, 0
40      @ 15, 0
41      IF ANS = 'N'
42        USE
43        RELEASE ALL
44        RETURN
45      ENDIF
46    ELSE
47      STORE ' ' TO ANS
48      @ 14, 0
49      @ 14,10 SAY 'Are you sure? (Y/N) ' GET ANS PICTURE '!'
50      READ
51      IF ANS = 'Y'
52        STORE T TO DONE
53      ENDIF
54    ENDIF
55  ENDDO
56  STORE ' ' TO ANS
57  DO WHILE .NOT. ANS $ 'NY'  * confirm to save the new item
58    STORE 'Y' TO ANS
59    @ 14,10 SAY 'Do you really want to save this item? (Y/N) 'GET ANS PICTURE '!'
60    READ
61  ENDDO
62  IF ANS = 'Y'
63    APPEND BLANK
64    REPLACE DESC WITH MDESC, ITEM:NO WITH MITEM:NO, PRICE WITH MPRICE;
65            ON:HAND WITH MON:HAND
66  ENDIF
67  USE
68  RELEASE ALL
```

列表 19

```
Listing of: B:DREPORT.CMD
1   * dreport.cmd
2   SET TALK OFF
3   ERASE
4   STORE '        ' TO FIDDATE
5   @ 10,10 SAY 'Enter the day to report? YY/MM/DD ' GET FIDDATE PICTURE '@R 99/99/99'
6   READ
7   SELECT SECONDARY
8   USE INVOICE INDEX INVOICE
9   FIND &FIDDATE
10  IF # = 0
11    STORE ' ' TO ANS
12    @ 10,10
13    @ 10,10 SAY 'No invoice in that day.Press any key to continue. ' GET ANS
14    READ
15    RETURN
16  ENDIF
17  STORE INVOICE TO LOWER
18  DO WHILE (.NOT. EOF) .AND. (DATE = FIDDATE)
19    SKIP
20  ENDDO
21  IF EOF
22    STORE INVOICE TO UPPER
23  ELSE
24    STORE INVOICE - 1 TO UPPER
25  ENDIF
26  SELECT PRIMARY
27  USE ITEM INDEX ITEM
28  TOTAL ON ITEM:NO TO DAILY FIELD TOTAL, QUANTITY;
29                    FOR (INVOICE )= LOWER) .AND. (INVOICE (= UPPER)
30  USE DAILY
31  UPDATE FROM MST:ITEM ON ITEM:NO REPLACE DESC
32  STORE $(FIDDATE,7,2)+'/'+$(FIDDATE,4,2)+'/'+$(FIDDATE,1,2) TO FIDDATE
33  STORE DATE() TO MDATE
34  SET DATE TO &FIDDATE
35  ERASE
36  REPORT FORM DAILY
37  STORE 'N' TO ANS
38  @ 23, 0 SAY 'Do you want to print on printer? (Y/N) ' GET ANS PICTURE '!'
39  READ
40  IF ANS = 'Y'
41    SET EJECT OFF
```

列表 17

Listing of: D:EDITITEM.CMD

```
* edititem.cmd
ERASE
@ 0,25 SAY 'E D I T   I T E M'
@ 0,60 SAY DATE()
@ 2, 0 SAY 'ITEM NO DESCRIPTION              PRICE    ON HAND'
USE MST:ITEM
STORE F TO DONE
DO WHILE .NOT. DONE  * loop control edit several items one time
  @ 3, 0 SAY '-------------------------------------';
           + '-------------------------------------'
  LIST NEXT 16 OFF
  @ 20, 0 SAY '-------------------------------------';
            + '-------------------------------------'
  IF EOF
    GOTO TOP
  ENDIF
  STORE ' ' TO ANS
  @ 21, 0 SAY '(E)nter item number, e(X)it or (space bar) to continue. ';
         GET ANS PICTURE '!'
  READ
  IF ANS = 'E'
    STORE T TO DONE
  ENDIF
  IF ANS = 'X'
    CLEAR
    RETURN
  ENDIF
ENDDO
STORE 0 TO NUMBER, NEWPRICE
STORE F TO DONE
DO WHILE .NOT. DONE    * repeat until all enter is valid
  @ 21, 0
  @ 21, 0 SAY 'Enter number  ' GET NUMBER PICTURE '999999'
  @ 21,40 SAY 'Enter new price ' GET NEWPRICE PICTURE '99999.99'
  READ
  GOTO BOTTOM
  IF (NUMBER ) 0) .OR. (NUMBER ( 1) .OR. (NEWPRICE ( 0)
    STORE ' ' TO MESS
    IF NEWPRICE ( 0
      STORE 'Invalid price' ' TO MESS
    ENDIF
    IF (NUMBER  ) 0) .OR. (NUMBER ( 1)
      STORE 'Item number not exist' ' + MESS TO MESS
    ENDIF
    @ 22, 0
    @ 22, 0 SAY MESS + 'Please re-enter!'
  ELSE
    GOTO NUMBER
    REPLACE PRICE WITH NEWPRICE
    STORE 'N' TO ANS
    @ 21, 0
    @ 22, 0
    @ 21, 10 SAY 'Do you want to edit more? (Y/N) ' GET ANS PICT '!'
    READ
    IF ANS = 'N'
      STORE T TO DONE
    ENDIF
  ENDIF
ENDDO
USE
RELEASE ALL
```

列表 18

Listing of: D:ONHAND.CMD

```
* onhand.cmd
ERASE
@ 0,25 SAY 'O N   H A N D   M E N U'
@ 0,60 SAY DATE()
@ 2, 0 SAY 'ITEM NO DESCRIPTION              PRICE   ON HAND'
USE MST:ITEM
STORE F TO DONE
DO WHILE .NOT. DONE
  @ 3, 0 SAY '-------------------------------------';
           + '-------------------------------------'
  LIST NEXT 16 OFF
  @ 20, 0 SAY '-------------------------------------';
            + '-------------------------------------'
  IF EOF
    GOTO TOP
  ENDIF
  STORE ' ' TO ANS
  @ 21, 0 SAY '(E)nter item number, e(X)it or (space bar) to continue. ';
         GET ANS PICTURE '!'
  READ
  IF ANS = 'X'
    CLEAR
    RETURN
  ENDIF
  IF ANS = 'E'
    STORE T TO DONE
  ENDIF
ENDDO
STORE 0 TO NUMBER, ADD
STORE F TO DONE
DO WHILE .NOT. DONE
  @ 21, 0
  @ 21, 0 SAY 'Enter item number ' GET NUMBER PICTURE '999999'
  @ 21,40 SAY 'Add on hand ' GET ADD PICTURE '99999999'
  READ
  GOTO BOTTOM            * check if item number and quantity valid
  IF (NUMBER ) 0) .OR. (NUMBER ( 1) .OR. (ADD ( 1)
    STORE ' ' TO MESS
    IF ADD ( 1
      STORE 'Invalid quantity! ' TO MESS
    ENDIF
    IF (NUMBER ) 0) .OR. (NUMBER ( 1)
      STORE 'Item number not exist' ' + MESS TO MESS
    ENDIF
    @ 22, 0
    @ 22, 0 SAY MESS + 'Please re-enter. '
  ELSE
    GOTO NUMBER
    REPLACE ON:HAND WITH ON:HAND + ADD
    STORE 'Y' TO ANS
    @ 21, 0
    @ 22, 0
    @ 21, 0 SAY 'Do you want to add more on hand? (Y/N) ' GET ANS PICT '!'
    READ
    IF ANS = 'N'
      STORE T TO DONE
    ENDIF
  ENDIF
ENDDO
USE
RELEASE ALL
```

列表 20

Listing of: B:PREPRINT.CMD

```
* preprint.cmd
SELECT SECONDARY
USE INVOICE
GOTO MINVOICE
SELECT PRIMARY
USE PRINT
DELETE ALL
PACK
APPEND FROM ITEM FOR INVOICE = MINVOICE
INDEX ON ITEM:NO TO PRINT
UPDATE ON ITEM:NO FROM MST:ITEM REPLACE DESC
REPLACE ALL PRICE WITH TOTAL/QUANTITY
SUM TOTAL TO RESULT
```

列表 21

Listing of: B:PRINT.CMD

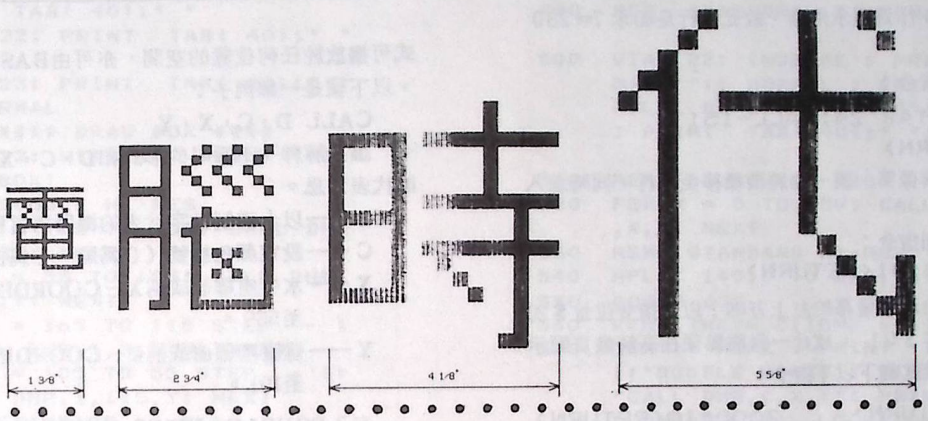
```
* print.cmd
ERASE
? '-------------------------------------------------------------------'
? 'INVOICE NUMBER   : ' + STR(MINVOICE,8,0)
? 'DATE             : ' + $(S.DATE,7,2) + '/' + $(S.DATE,4,3) + $(S.DATE,1,2)
? 'SOLD BY          : ' + S.SOLD:BY
? '-------------------------------------------------------------------'
? 'ITEM NO DESCRIPTION          QUANTITY      PRICE       TOTAL'
? '-------------------------------------------------------------------'
LIST ITEM:NO, DESC, QUANTITY, PRICE, TOTAL  ALL OFF
? '-------------------------------------------------------------------'
? ' '
?                                          TOTAL : ' + STR(RESULT,10,2)
```

中文咭
應用專欄

BISCUIT

適用於各種印字機
可印出大細不同中文字
改良版中文大字海報編印器

大小不同的中文字



相信擁有KX－P1091或EN1091這類印字機的朋友，在執行23期的「中文大字海報編印器」的時候，都有以下的經歷——印出來的字體少了一截。細查究竟，發覺原程式是設計給CP－80這類印字機使用，而1091每行所印的點數較CP－80爲少（請參看22期的活用印字機及PRINTSHOP助手），於是每行都印少了幾個黑色的方格，形成字體少了一截。

細讀原程式，發覺在印字方面，還有改良的地方。首先是印字機的相容問題，其次是字體的大小控制，最好能將不同大小的字體放在海報中央；或將大字放在中央，小字放在下面做備註。於是，筆者作了以下的嘗試：

第一步，鍵入程式（一），然後加上以下兩行指令（在DOS 3・3系統下）：

```
1  D$=CHR$(4):F$="IMPROVE BIG CHR$"
   :POKE 33,33
2  PRINT D$"OPEN"F$:PRINT D$"WRITE"
   F$:POKE 33,33:LIST 32,630:PRINT D$
   "CLOSE":POKE 33,40:END
```

第二步，執行上述程式，產生一個以IMPROVE BIG CHR$爲名的TEXT FILE。

第三步，LOAD 23期的BIG CHR$，跟著EXEC上述的TEXT FILE。最後執行程式。

在程式的執行過程中，加入了兩項功能。第一是字體的大小，1091有3種選擇，CP－80有5種（1－5，1最細，5最大）。其次是將字體放在中央的選擇。還有，此程式是設計給大部份與EPSON指令相容的印字機。行號32的MARGIN設定值是給1091使用的，CP－80的MARGIN值是80。其他的印字機可將它在CCDOS系統下每行所能印英文字數的最大值代入MARGIN值內。

若想字體放在中央，你的MARGIN值必需是最大值，否則字體會移位。在不需將字體放在中央的情況下，MAR－GIN值祇影响字體的大小。

最後，還有一點值得法意的，若大家的中文咭不是放在SLOT 4或中文系統是使用HGR1，記得將行號7的CD值或HG值更改，否則程式便不能正常運行。

購有本期程式磁碟的讀者，我們經已將一個名爲「IM－PROVE BIG CHR$」的TEXT FILE放在磁碟上。不過只要按上文所述的步驟，便可印出大小不同的中文字海報了。■

程式1

```
32  LET MARGIN = 60:BIG =  INT (MARGIN / 16)
34  DIM SPACE$(MARGIN - 16)
36  LET SPACE$(0) = ""
38  FOR I = 1 TO MARGIN - 16:SPACE$(I) = SPACE$(I - 1) + " "
: NEXT I
40  HOME : PRINT "SIZE (1-";BIG;") : "
42  VTAB 1: HTAB 14: INPUT "";SIZE
44  IF SIZE < 1 OR SIZE > BIG THEN 40
46  PRINT : PRINT
48  INPUT "CENTRE TEXT (Y/N) ? ";CENTRE$
50  IF CENTRE$ <  > "Y" AND CENTRE$ <  > "N" THEN 48
501  POKE 2040 + CD,MARGIN
517  FOR T = 1 TO SIZE
518  IF CENTRE$ = "Y" THEN  PRINT SPACE$((MARGIN - 16 * SIZE
) / 2);
540  PRINT SPACE$(SIZE);
605  FOR T = 1 TO SIZE
608  IF CENTRE$ = "Y" THEN  PRINT SPACE$((MARGIN - 16 * SIZE
) / 2);
630  PRINT SPACE$(SIZE);
```

並非IIe的專利

II十上的雙倍高解像密度

〈邵英傑〉

蘋果用家都知道，它的高解像度是280×192點，不過，祇要你有一部II PLUS，便可得到一幅由560×192點的解像，用黑白兩種顏色顯示。

首先，我們不妨看看蘋果電腦的高解像頁是如何組成，它總共有兩高解像頁，第一頁由記憶位址$2000至$3FFF組成，第二頁由記憶位址$4000至$5FFF組成，每頁可分爲192行，每行由40BYTES記憶體結合而成，而每BYTE只用其中7 BITS作爲顯示用途，故此每行是40*7=280點。

請鍵入下列指令：

```
HGR:VTAB 24:CALL-151
<RETURN>
```

這會顯示高解像第一頁，並將游標移至底行，同時進入監督程式。

請再鍵入下列指令：

```
2000:01<RETURN>
```

看到有一點出現在螢幕的左上方嗎？以上指令位址$2000的內含值設爲#$01，故此一對應點便在高解像頁顯示出來了，現在請試試鍵下以下指令：

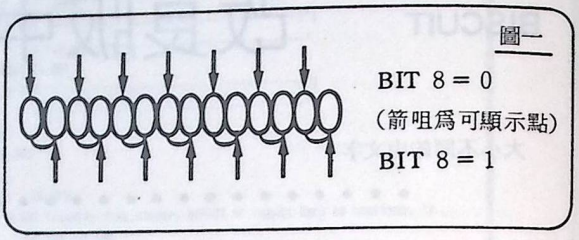
```
2000:02<RETURN>     2000:10<RETURN>
2000:04<RETURN>     2000:20<RETURN>
2000:08<RETURN>     2000:40<RETURN>
```

現在你們知道只要在適當的位置位址設上不同的值，便可以得到不同位置的點了，我們每BYTE只取用7 BITS，那麼還有1 BIT是作那種用途呢？其實最後1 BIT是用了作COLOR BIT，它是決定顏色的位元，同時你亦可利用它來顯示雙倍高解像頁。實際上，其餘7 BITS都是對應兩點，分左右兩點，若第8 BIT是0，那麼7 BITS便用左面一點顯示；否，則用右面一點顯示（請參看圖一）。請再鍵入下列指令：

```
2000:01<RETURN>     2000:88<RETURN>
2000:81<RETURN>     2000:10<RETURN>
2000:02<RETURN>     2000:90<RETURN>
2000:82<RETURN>     2000:20<RETURN>
2000:04<RETURN>     2000:A0<RETURN>
2000:84<RETURN>     2000:40<RETURN>
2000:08<RETURN>     2000:C0<RETURN>
```

你會發覺螢幕上的點是有很少的移動，這就是因爲BIT 8改變所造成的。細心觀察，既然每BIT都有兩個對應點，那麼7 BITS便有14個對應點，亦即是說，一行可以有560個對應點。不過你同時亦會發覺在APPLE II PLUS不可以同時顯示560點，若將BIT 8改變，另外7 BITS亦同時改變，換句話說你不能要求在同1 BYTE，有些BIT用左點顯示，有些BIT用右點顯示，這是絕對不成的。

爲充份使用高解像頁，我編寫了一個機械碼程式，此程式可擺放於任何位置的空間，亦可由BASIC程式呼叫使用，以下便是一個例子：



圖一

```
CALL D,C,X,Y
```

讓我解釋一下呼叫的形式和D，C，X，Y這四個參數的代表意思。

D——以十進制數字代表的機器語言程式的起始位址。

C——設定顏色旗號（0爲黑，1爲白）。

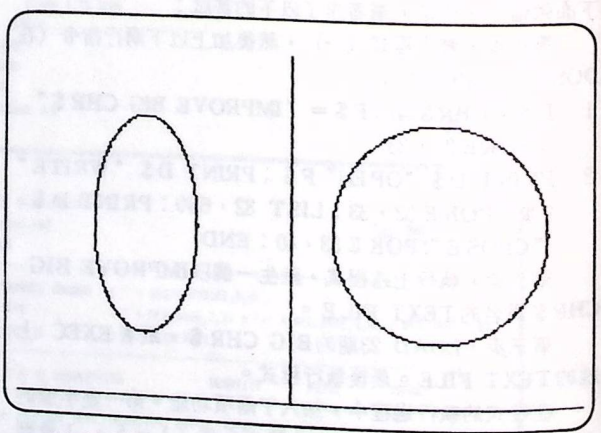
X——水平座標值或稱X－COORDINATE，其值爲0至559。

Y——垂直座標值或稱Y－COORDINATE，其值爲0至191。

存入DOUBLE・HIRES檔案的方法很簡單，只須依照LISTING 1程式鍵入，以BSAVE DOUBLE・HIRES，A$300，L$2E便可，至於LISTING 2的DHR・DEMO祇是一個示範程式，可有可無，並不需要一定存入檔案。

爲了解程式的使用法，你們不妨嘗試鍵入DHR・DEMO，看看此程式在高解像頁的運作情況。DHR・DEMO會將一個50×50點的長方形顯示出來，同時，有一個圓形顯示，你不妨比較一下它是否與一般的高解像頁的顯示一樣。

本文兩個程式同時收錄在同期出版的程式磁碟上。名字分別爲：機械碼程式DOUBLE HIRES及APPLESOFT程式DHR・DEMO。



圖(2)雙倍密度圖像（左邊）與單密度的圖像（右邊）的比較，彼此都有同一半徑。

II十上的雙倍高解像密度

```
1  REM **********************
2  REM *       DHR.DEMO       *
4  REM * COPYRIGHT (C) 1985 *
6  REM *     BY COMPUHOUSE    *
7  REM **********************
50  REM  INITIALIZE AND SET UP HG
     R SCREEN
60  HOME : HGR : HCOLOR= 3: PRINT

70  PRINT  CHR$ (4);"BLOAD DOUBLE
     .HIRES"
80 DHR = 768: REM  ADDRESS OF
     DOUBLE HI-RES ROUTINE
90  HPLOT 140,0 TO 140,159: HPLOT
     0,159 TO 279,159
100  INVERSE : VTAB 21: PRINT "
      DOUBLE HIRES                HI
     RES"; TAB( 40);" "
110  VTAB 22: PRINT  TAB( 40);" "
     ;
120  VTAB 23: PRINT  TAB( 40);" "
     ;: NORMAL
130  REM  **** DRAW BOX ****
140  VTAB 22: HTAB 15: PRINT "50
     X 50 BOX"
150  REM  DOUBLE HI-RES
160  FOR X = 115 TO 165: CALL DHR
     ,1,X,55: NEXT
170  FOR Y = 55 TO 105: CALL DHR,
     1,165,Y: NEXT
180  FOR X = 165 TO 115 STEP  - 1
     : CALL DHR,1,X,105: NEXT
190  FOR Y = 105 TO 55 STEP  - 1:
      CALL DHR,1,115,Y: NEXT
200  REM  STANDARD HI-RES
210  FOR X = 185 TO 235: HPLOT X,
     55: NEXT
220  FOR Y = 55 TO 105: HPLOT 235
     ,Y: NEXT
230  FOR X = 235 TO 185 STEP  - 1
     : HPLOT X,105: NEXT
240  FOR Y = 105 TO 55 STEP  - 1:
      HPLOT 185,Y: NEXT
250  GOSUB 630: GOSUB 610: REM
     WAIT FOR KEYSTROKE AND
      DIVIDE SCREEN
260  REM  **** DRAW CIRCLE ****
270  VTAB 22: HTAB 12: PRINT "CIR
     CLE RADIUS 50"
280  REM  DOUBLE HI-RES
290 XC = 140:YC = 80:R = 50:PA =
     0:PB = 6.28318:DP = .0174532
     778
300  FOR P = PA TO PB STEP DP:X =
     R *  COS (P):Y = R *  SIN (P
     ):X = XC + X:Y = Y + YC: CALL
     DHR,1,X,Y: NEXT
310  REM  STANDARD HI-RES
320  HCOLOR= 3
330 XC = 210:YC = 80:R = 50:PA =
     0:PB = 6.28318:DP = .0174532
     778
340  FOR P = PA TO PB STEP DP:X =
     R *  COS (P):Y = R *  SIN (P
     ):X = XC + X:Y = Y + YC: HPLOT
     X,Y: NEXT
350  GOSUB 630: GOSUB 610
360  REM  **** DRAW SINE WAVE ***
     *
370  VTAB 22: HTAB 1: INVERSE : PRINT
      TAB( 15);" ";: NORMAL : PRINT
     "SINE WAVE";: INVERSE : PRINT
      TAB( 40);" ";: NORMAL
380  REM  DOUBLE HI-RES
390  FOR A = 0 TO 278
400 X = (A - 140) / 38:Y =  SIN (
     X):YP = 96 - (Y * 30): IF YP
      < 0 AND YP > 191 THEN 420
410  CALL DHR,1,A,YP
420  NEXT
430  REM  STANDARD HI-RES
440  FOR A = 140 TO 278
450 X = (A - 210) / 19:Y =  SIN (
     X):YP = 96 - (Y * 30): IF YP
      < 0 AND YP > 191 THEN 470
460  HPLOT A,YP
470  NEXT
480  GOSUB 630: GOSUB 610
490  REM  **** DRAW DIAGONAL ****

500  VTAB 22: INVERSE : PRINT  TAB(
     8);" ";: NORMAL : PRINT "PAR
     ALLEL DIAGONAL LINES";: INVERSE
     : PRINT  TAB( 40);" ";: NORMAL

510  REM  DOUBLE HI-RES
520  FOR X = 0 TO 159: CALL DHR,1
     ,X,X: NEXT
530  REM  STANDARD HI-RES
540  HPLOT 140,0 TO 220,159
550  GOSUB 630
560  REM  INSTRUCTIONS *****
570  TEXT : HOME : PRINT  TAB( 14
     );"DOUBLE HIRES": VTAB 3: PRINT
     "CALL DHR,C,X,Y": PRINT : PRINT
     " DHR=DECIMAL LOCATION OF DO
     UBLE HI-RES        ROUTINE"
580  PRINT : PRINT " C=COLOR (0=B
     LACK,1=WHITE)": PRINT : PRINT
     " X=X COORD. RANGE (0 TO 559
     )": PRINT : PRINT " Y=Y COOR
     D. RANGE (0 TO 191)": VTAB 2
     0: PRINT " END OF DEMO"
590  END
600  REM  SUBROUTINE TO DIVIDE
     SCREEN
610  HGR : HCOLOR= 3: HPLOT 140,0
      TO 140,159: HPLOT 0,159 TO
     279,159: RETURN
620  REM  SUBROUTINE TO WAIT FOR
     RETURN TO BE PRESSED
630  VTAB 24: PRINT  TAB( 13);"PR
     ESS <RETURN>";
640 X =  PEEK ( - 16384): IF X <
     128 THEN 640
650  POKE  - 16368,0
660  IF X <  > 141 THEN 640
670  VTAB 24: HTAB 1: CALL  - 958

680  RETURN


*****************
* DOUBLE.HIRES *
*  A$300.L$2E  *
*****************
0300- A9 00 85 E4 20 4C E7 E0
0308- 00 F0 04 A9 7F 85 E4 20
0310- BE DE 20 67 DD 20 52 E7
0318- 18 66 51 66 50 90 08 D8
0320- 18 A5 E4 69 80 85 E4 20
0328- BF F6 20 57 F4 60 00
```

趨吉避凶的生理周期分析預報系統

大家有沒有發覺有時就算有充足的睡眠，晨早起來亦會覺得精神不振，整天工作也會提不起勁？

根據多年研究所得出之生理周期理論，人類的體力、情緒以及智力均作周期性的變化。它們的周期分別爲23，28和33日，變化起伏就如正弦曲線（SINE CURVE）一樣。每個周期的開始和終結稱爲極端日子，整個人就會陷入低潮。各人所受的影响會隨着健康、年齡、性格而有所不同，但是無論如何，知道自己每日之生理狀態總是件好事。

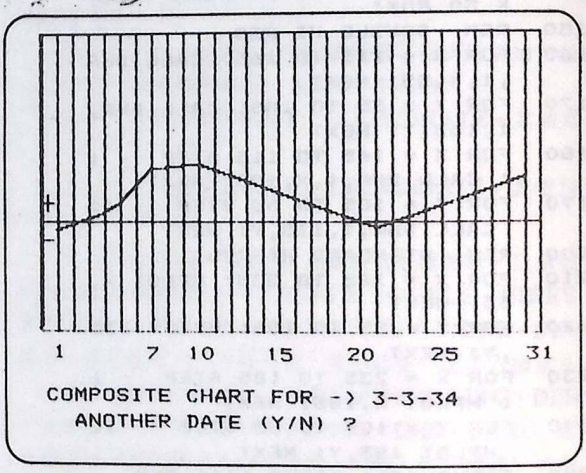
本文之APPLESOFT程式COMP・BIO能自動替你計算任何一日之體力、情緒及智力的狀態，而且更會以圖表形式顯示出來。自己鍵入程式的讀者可照着列表打入，再用SAVE COMP・BIO便能把程式存在碟中。購有本期程式磁碟的讀者，只要RUN COMP・BIO即可。

由於個人的性向與生理狀態有莫大的關係，以下筆者提供了一個小小的心理測驗來測定各位的性向，使生理狀態的推算更爲準確。請逐一回答以下的各個問題及用紙把答案記下，但不要花太多時間思考。

1. 面對着危險，你會
   a. 保持鎮靜
   b. 驚恐哭泣
   c. 盡力解決
2. 你喜歡
   a. 玩需要思考的遊戲
   b. 看電影
   c. 參與體育活動
3. 當你要作出一項抉擇，你會
   a. 小心選擇
   b. 看心情而定
   c. 嘗試每個選擇
4. 有沒有朋友認爲你
   a. 有邏輯
   b. 有衝勁
   c. 堅強

```
PHYSICAL CYCLE              469  COMPLETED
DAYS INTO CURRENT CYCLE: 23   LOW

=======================================
EMOTIONAL CYCLE             385  COMPLETED
DAYS INTO CURRENT CYCLE: 7    HIGH

=======================================
INTELLECTUAL CYCLE          326  COMPLETED
DAYS INTO CURRENT CYCLE: 29   LOW

=======================================

COMPOSITE CHART (Y/N) ?
```

圖(1) 閣下處於某日的生理周期情況。



圖(2) 綜合體力，情緒及智力三條曲線得出來的曲線圖

5. 你的日常活動屬於
   a. 需要不斷分析
   b. 藝術性
   c. 戶外活動
6. 你認爲你的性格是
   a. 理性
   b. 感性
   c. 外向
7. 假如你有適當的資格，你希望成爲
   a. 科學家
   b. 藝術表演者
   c. 體育家
8. 如果你要處罰一個孩子，你會
   a. 選擇一個相關的刑罰
   b. 儘量避免
   c. 讓他解釋
9. 你會在空餘時間
   a. 看書
   b. 看電視
   c. 找工作做

現在就數數a，b和c的數目，假如某項的數目多於5，你就是該型，否則就是平均型：

a —INTELLECTUAL
b —EMOTIONAL
c —PHYSICAL

趨吉避凶的生理周期分析預報系統

一般的生理周期計算程式（如DOS 3·3 SYSTEM MASTER 上的那個，以整數BASIC 編寫）都只是顯示一段時期內的三條曲線，你需要花時間來理解對你整體的影响。而本程式能自動製作出一條組合曲線，你便可一目了然地知道每日的生理狀態。

至於此程式的操作則是十分簡單，初學者一看便會，只須一一輸入名字，出生日期，性向（參看心理測驗的結果），及目標日期便可。此程式有兩個主要的功能，第一是計算並顯示出智力、情緒及體力的狀態，你可根據得出的資料參考表一而決定當天該做什麽（別太認眞！）；有需要之時可再替你繪出整體生理狀態的曲線以作參考，即本程式之第二項功能。

有一點要補充的是在程式列表第860－870 行內，筆者加上了自己及家人的名字和出生日期（註BMO ＝出生月份，BDA＝出生之日子，BYR＝出生年份）。各位可把其更動而使到每次執行程式之時，不必再打入出生日期，因程式會根據名字而找出來。 ▣

表一

周期狀態			建議之活動
體力	情緒	智力	
高	高	高	計劃及實行之最佳時間
高	高	低	適合體力活動及與人相處
高	低	低	情緒低落，思考不靈，只宜體育運動
低	低	低	休息或做不重要的事
低	低	高	最好的計劃時間，避免無謂的活動
低	高	低	花時間思考以解決問題
高	低	高	避免與人發生磨擦

```
10  REM  ************************
20  REM  *       COMP.BIO        *
30  REM  *    BY COMPUHOUSE      *
40  REM  *  COPYRIGHT(C) 1985   *
50  REM  *  BY COMPUTING AGE    *
60  REM  ************************
80  IF  PEEK (104) <  > 64 THEN  POKE
    103,1: POKE 104,64: POKE 163
    84,0: PRINT  CHR$ (4)"RUN CO
    MP.BIO"
90  TEXT : HOME
100  REM  DIMENSION ARRAYS
110 :
120  DIM D(8),CIN(33,2),CEM(28,2)
    ,CPH(23,2),CV(31),PL(31),MA(
    12),MN(12)
130  REM MA = DAYS PAST, MN = DAY
    S IN MONTHS
140  FOR I = 1 TO 12: READ MA(I),
    MN(I): NEXT
150 :
160  REM READ INTELLECTUAL VALUE
170 :
180  FOR I = 1 TO 33: FOR V = 1 TO
    2: READ CIN(I,V): NEXT : NEXT

190 :
200  REM  READ EMOTIONAL VALUES
210 :
220  FOR I = 1 TO 28: FOR V = 1 TO
    2: READ CEM(I,V): NEXT : NEXT

230 :
240  REM READ PHYSICAL VALUES
250 :
260  FOR I = 1 TO 23: FOR V = 1 TO
    2: READ CPH(I,V): NEXT : NEXT

270  GOTO 800
280  DATA   0,31,31,28,59,31,90,3
    0,120,31,151,30,181,31,212,3
    1,242,30,273,31,304,30,334,3
    1
290 :
300  REM  COMPOSITE VALUES - INT
310 :
320  DATA 15,3,18,6,21,9,24,12,27
    ,15,30,18,33,21,36,24,34,22,
    31,19,28,16,25,13,22,10,19,7
    ,16,4,13,1,10,-2,7,-5,4,-8,1
    ,-11,-2,-14,-5,-17,-8,-20,-1
    1,-23,-12,-24,-9,-21,-6,-18,
    -3,-15,0,-12,3,-9,6,-6,9,-3,
    12,0
330 :
340  REM COMPOSITE VALUES - EMO
350 :
360  DATA  14,4,18,8,22,12,26,16,
    30,20,34,24,38,28,34,24,30,2
    0,26,16,22,12,18,8,14,4,10,0
    ,6,-4,2,-8,-2,-12,-6,-16,-10
    ,-20,-14,-24,-18,-28,-14,-24
    ,-10,-20,-6,-16,-2,-12,3,-8,
    6,-4,1,0
370 :
380  REM COMPOSITE VALUES - PHY
390 :
400  DATA  10,2,12,4,14,6,16,8,18
    ,10,19,11,17,9,15,7,13,5,11,
    3,9,1,7,-1,5,-3,3,-5,1,-7,-1
    ,-9,-3,-11,-2,-10,0,-8,2,-6,
    4,-4,6,-2,8,0
410 :
420  REM ASSIGN COMPOSITE VALUE
430 :
440 D1 = (DZS / 33) -  INT (DZS /
    33):D2 = D1 * 33:MI(1) =  INT
    (D2 + .5): IF MI(1) = 0 THEN
    MI(1) = 1
450 D3 = (DZS / 28) -  INT (DZS /
    28):D4 = D3 * 28:EI(1) =  INT
    (D4 + .5): IF EI(1) = 0 THEN
    EI(1) = 1
```

```
460 D5 = (DZS / 23) -  INT (DZS /
    23):D6 = D5 * 23:PI(1) =  INT
    (D6 + .5): IF PI(1) = 0 THEN
    PI(1) = 1
470 FOR I = 1 TO MN(TMN)
480 IF TP$ = "INT" THEN CV(I) =
    CIN(MI(1),1) + CEM(EI(1),2) +
    CPH(PI(1),2): GOSUB 1500: NEXT

490 IF TP$ = "EMO" THEN CV(I) =
    CIN(MI(1),2) + CEM(EI(1),1) +
    CPH(PI(1),2): GOSUB 1500: NEXT

500 IF TP$ = "PHY" THEN CV(I) =
    CIN(MI(1),2) + CEM(EI(1),2) +
    CPH(PI(1),1): GOSUB 1500: NEXT

510 IF TP$ = "AVG" THEN CV(I) =
    CIN(MI(1),2) + CEM(EI(1),2) +
    CPH(PI(1),2): GOSUB 1500: NEXT

520 :
530 REM   MODIFY PLOT VALUES
540 :
550 FOR I = 1 TO MN(TMN)
560 IF CV(I) = 0 THEN PL(I) = 10
    0: NEXT : GOTO 630
570 IF CV(I) > 0 AND CV(I) < 100
     THEN PL(I) = 100 - CV(I): NEXT
    : GOTO 630
580 IF CV(I) > 100 AND CV(I) < 1
    60 THEN PL(I) = CV(I): NEXT
    : GOTO 630
590 IF CV(I) < 0 THEN PL(I) = (C
    V(I) *  - 1) + 100: NEXT
600 :
610 REM  FRAME COMPOSITE CHART
620 :
630 HGR : HCOLOR= 3:P = 1
640 HPLOT 0,0 TO 279,0 TO 279,15
    9 TO 0,159 TO 0,0: HPLOT 0,1
    00 TO 279,100
650 FOR I = 1 TO 279 STEP 9: HPLOT
    I,0 TO I,159: NEXT
660 HPLOT 1,90 TO 7,90: HPLOT 4,
    86 TO 4,94: HPLOT 2,110 TO 7
    ,110
670 HOME : VTAB 21: PRINT " 1
       7    10     15      20      2
    5     31": POKE 34,22
680 :
690 REM    PLOT COMPOSITE GRAPH

700 :
710 P = 1
720 HPLOT 9,PL(P)
730 P = 2
740 FOR I = 18 TO 279 STEP 9
750 HPLOT  TO I,PL(P): VTAB 22: IF
    P = MN(TMN) THEN 1350
760 P = P + 1: NEXT : GOTO 1350
770 :
780 REM   OPTIONS AVAILABLE
790 :
800 TEXT : HOME : HTAB 9: INVERSE
    : PRINT " B I O R H Y T H M
    S ": NORMAL : PRINT  TAB( 9)
    "----------------------": POKE
    34,2: PRINT : PRINT "** COPY
    RIGHT 1985 BY COMPUTING AGE
     **"
810 VTAB 8: HTAB 5: PRINT "1) PO
    SITION IN CYCLE ": VTAB 10: HTAB
    5: PRINT "2) COMPOSITE CHART
     ": VTAB 12: HTAB 5: PRINT "
    3) QUIT "
820 VTAB 21: PRINT "ENTER NUMBER
     OF CHOICE--> ";: GET N$:N =
     VAL (N$): IF N < 1 OR N > 3
     THEN  HOME : VTAB 12: GOTO
    840
830 GOTO 850
840 FOR I = 1 TO 30:X =  PEEK ( -
    16336) +  PEEK ( - 16336) +
     PEEK ( - 16336): NEXT : HTAB
    12: FLASH : PRINT " B A D
    C H O I C E ": FOR I = 1 TO
    1000: NEXT : HOME : GOTO 800

850 IF N = 3 THEN 1370
860 HOME : VTAB 12: INPUT "ENTER
     YOUR FIRST NAME PLEASE ->";
    NZ$: IF NZ$ = "LEE" THEN BMN
     = 9:BDA = 24:BYR = 48: PRINT
    : PRINT : GOTO 920
870 IF NZ$ = "BRENDA" THEN BMN =
    8:BDA = 14:BYR = 53: PRINT :
     PRINT : GOTO 920
880 HOME : VTAB 5: PRINT "ENTER
    MONTH,DAY & YEAR OF BIRTH": PRINT

890 VTAB 10: CALL  - 958: VTAB 1
    1: PRINT "(1=JAN, 2=FEB, ETC
    .)": VTAB 10: INPUT "MONTH->
    ";K$:BMN =  VAL (K$): IF BMN
     < 1 OR BMN > 12 THEN 890
900 VTAB 13: CALL  - 958: INPUT
    "DAY->";K$:BDA =  VAL (K$): IF
    BDA < 1 OR BDA > 31 THEN 900

910 VTAB 15: CALL  - 958: VTAB 1
    6: PRINT "(LAST 2 DIGITS)": VTAB
    15: INPUT "YEAR->";K$:BYR =
     INT ( VAL (K$)): IF BYR < 0
     OR BYR > 99 THEN 910
920 POKE 34,0: HOME : HTAB 11: INVERSE
    : PRINT " PERSONALITY TYPES ": NORMAL
    : PRINT "-------------------
    ----------------------": PRINT
    : PRINT : REM  40 HYPHENS
930 VTAB 6: PRINT "INTELLECTUAL
    = INT": PRINT : HTAB 4: PRINT
    "EMOTIONAL = EMO": PRINT : HTAB
    5: PRINT "PHYSICAL = PHY": PRINT
    : HTAB 6: PRINT "AVERAGE = A
    VG": PRINT : PRINT : INPUT "
    ENTER YOUR TYPE -> ";TP$
940 IF TP$ <  > "INT" AND TP$ <
     > "EMO" AND TP$ <  > "PHY" AND
    TP$ <  > "AVG" THEN  HOME : VTAB
    10: GOTO 920
950 :
960 REM   INPUT TARGET DATE
970 :
```

趨吉避凶的生理周期分析預報系統

```
980  HOME : PRINT "ENTER TARGET D
     ATE...."
990  VTAB 10: CALL  - 958: VTAB 1
     1: PRINT "(1=JAN, 2=FEB, ETC
     .)": VTAB 10: INPUT "MONTH->
     ";K$:TMN =  VAL (K$): IF TMN
      < 1 OR TMN > 12 THEN 990
1000  VTAB 13: CALL  - 958: INPUT
     "DAY->";K$:TDA =  VAL (K$): IF
     TDA < 1 OR TDA > 31 THEN 100
     0
1010  VTAB 15: CALL  - 958: VTAB
     16: PRINT "(LAST 2 DIGITS)":
      VTAB 15: INPUT "YEAR->";K$:
     TYR =  INT ( VAL (K$)): IF T
     YR < 0 OR TYR > 99 THEN 1010

1020 :
1030  REM      COMPUTE DAYS
1040 :
1050  IF TYR < BYR THEN 870
1060 D = 0:T1Y = TYR:T2M = TMN:T3
     D = TDA: GOSUB 1100:TTD = D:
      REM  COMPUTE TOTAL PAST DAY
     S FROM 1900 TO TARGET DAY
1070 D = 0:T1Y = BYR:T2M = BMN:T3
     D = BDA: GOSUB 1100: REM COM
     PUTE DAYS FROM 1900 TO BIRTH
     DATE
1080 DAZ = TTD - D: GOTO 1140: REM
     TOTAL DAYS IN CYCLE IS TOTAL
      DAYS TO TARGET - TOTAL DAYS
      TO BIRTH
1100  FOR I = 0 TO T1Y:D = D + 36
     5: IF I / 4 =  INT (I / 4) THEN
     D = D + 1
1110  NEXT I:D = D + MA(T2M):D =
     D + T3D: IF T2M > 2 AND T1Y /
     4 =  INT (T1Y / 4) THEN D =
     D + 1
1120  RETURN
1130 :
1140  IF DAZ > 31 THEN DZS = DAZ -
     TDA
1150 PH =  INT (DAZ / 23):PI =  INT
     (DAZ - (PH * 23)):EM =  INT
     (DAZ / 28):EI =  INT (DAZ -
     (EM * 28)):ME =  INT (DAZ /
     33):MI =  INT (DAZ - (ME * 3
     3))
1160  IF MI = 0 THEN MI = 33
1170  IF MI < 1 THEN MI = 1
1180  IF EI = 0 THEN EI = 28
1190  IF EI < 1 THEN EI = 1
1200  IF PI = 0 THEN PI = 23
1210  IF PI < 1 THEN PI = 1
1215  IF DAZ = 0 THEN PI = 0:MI =
     0:EI = 0
1220  IF N = 2 THEN  POKE 34,0: HOME
     : VTAB 12: HTAB 18: FLASH : PRINT
     "WORKING": NORMAL : GOTO 440

1230 :
1240  REM DISPLAY RESULTS
1250 :
1260  HOME : HTAB 9: INVERSE : PRINT
     " B I O R H Y T H M S ": NORMAL
     : PRINT  TAB( 9)"----------
     ----------": POKE 34,2: VTAB
     5: REM 21 HYPHENS
1270  PRINT "PHYSICAL CYCLE
         "PH;: HTAB 30: PRINT "CO
     MPLETED": GOSUB 1410: PRINT
     "DAYS INTO CURRENT CYCLE: "P
     I"    "PN$: PRINT : GOSUB 154
     0
1280  PRINT "EMOTIONAL CYCLE
         "EM;: HTAB 30: PRINT "CO
     MPLETED": GOSUB 1440: PRINT
     "DAYS INTO CURRENT CYCLE: "E
     I"    "EN$: PRINT : GOSUB 154
     0
1290  PRINT "INTELLECTUAL CYCLE
         "ME;: HTAB 30: PRINT "CO
     MPLETED": GOSUB 1470: PRINT
     "DAYS INTO CURRENT CYCLE: "M
     I"    "MN$: PRINT : GOSUB 154
     0
1300 :
1310  REM       ENDING  ROUTINE
1320 :
1330  VTAB 23: PRINT "  COMPOSITE
      CHART (Y/N) ? ";: GET A$: PRINT
     A$: IF A$ = "Y" THEN  POKE 3
     4,0: HOME : VTAB 12: HTAB 18
     : FLASH : PRINT "WORKING": NORMAL
     : GOTO 440
1340  GOTO 1360
1350  HOME : VTAB 22: HTAB 5: PRINT
     "COMPOSITE CHART FOR -> "TMN
     "-"TDA"-"TYR
1360  VTAB 23: CALL  - 958: PRINT
     "       ANOTHER DATE (Y/N) ?
     ";: GET A$: IF A$ = "Y" THEN
      TEXT : HOME : GOTO 980
1370  TEXT : HOME : VTAB 12: PRINT
     "SEE YOU LATER THEN....": END

1380 :
1390  REM DECIDE STATUS: HIGH-LOW
     -CRITICAL
1400 :
1410  IF PI = 11 OR PI = 1 THEN P
     N$ = "CRITICAL": RETURN
1420  IF PI > 11 THEN PN$ = "LOW"
     : RETURN
1430 PN$ = "HIGH": RETURN
1440  IF EI = 14 OR EI = 1 THEN E
     N$ = "CRITICAL": RETURN
1450  IF EI > 14 THEN EN$ = "LOW"
     : RETURN
1460 EN$ = "HIGH": RETURN
1470  IF MI = 16 OR MI = 1 THEN M
     N$ = "CRITICAL": RETURN
1480  IF MI > 16 THEN MN$ = "LOW"
     : RETURN
1490 MN$ = "HIGH": RETURN
1500 MI(1) = MI(1) + 1: IF MI(1) >
     33 THEN MI(1) = 1
1510 EI(1) = EI(1) + 1: IF EI(1) >
     28 THEN EI(1) = 1
1520 PI(1) = PI(1) + 1: IF PI(1) >
     23 THEN PI(1) = 1
1530  RETURN
1540  FOR I = 1 TO 38: PRINT "=";
     : NEXT : PRINT : RETURN
```

創新多頁高解像螢幕模擬意念

文字及圖案編輯處理器

FONTRIX

市面上有售的WORD PROCESSOR，相信不下數十多種，而其中却以MICRO PRO 的WORD STAR 最爲有特色和高效能。

WORDSTAR 之所以成功，其最大原因爲它的特殊編印功能和強勁的編輯功能。一些簡易的數學符號如2的2次方符號也可編印出來，但相反一些比較複雜的符號如微積分算符，WORDSTAR 便有所欠缺了，這也是一般運行於APPLE Ⅱ上的文字處理器所沒法達到的！

在以往你們有沒有構思將APPLE 的高解像畫面，來用作文字與圖案編輯的工具呢！若能同具多款文字與圖案編輯的功能，這才是一個強勁和理想的文字處理器了。一如APPLE 32BIT 系列MACINTOSH的MACWRITE 文字處理系統，便包含了以上所提及的要求。

但你們想擁有一部APPLE 的MACINTOSH 系統，是必須付出一定的代價呢！

現在，美國DATA TRANSFORMS INC. 軟件公司便在APPLE Ⅱ上發展了一個高解像文字與圖畫的處理系統，其名稱爲FONTRIX。

顧名思義，FONTRIX是一個具有字形製造與圖案混合編輯功能的文字處理系統。假若你需要在文字報告中加入一些統計圖表，那麼FONTRIX 便可助你一臂之力了！

系統

整個FONTRIX系統是儲存於一張磁碟之上，其中是包括了隨系統預設的多款英文字體與符號圖案等。

單單利用預設的多種英文字體，便足以提供處理及編輯一篇極美化的文字印本了。

FONTRIX 系統基本是由APPLESOFT BASIC 所編寫的，而系統的主要心臟便是圖案與文字處理的子程式——GRAPHIC WRITER。由6502 機械言語所編寫而成的GRAPHIC WRITER，其運行速度和編輯功能是極爲強勁。

功能

FONTRIX 的主功能選表是可分列如下：

(I) INTRODUCTION & DEMONSTRATION
(N) NEW CONFIG URATION
(D) DISK ACCESS
(F) FONT EDITOR
(G) GRAPHIC WRITER
(P) PRINT GRAPHICS
(Q) QUIT FONTRIX SYSTEM

若你按下(I)鍵，FONTRIX 便會自動執行整個文字系統的輸入與操作示範的模擬，可謂生動而有趣，這好讓一個電腦門外漢也能概括地認識整個FONTRIX 系統。

(N)鍵功能是提供用者選定或改變FONTRIX的系統參數，以配合不同牌子的點陣印字機與操作介面。

FONTRIX 是可在單一，2部或以上的APPLE 磁碟系統上運行，所以(D)鍵功能便是讓用者置定FONTRIX 與系統文字檔案的磁碟機編號，以及一些磁碟檔案處理功能等。

造字

若按(F)鍵，FONTRIX 便會進入造字功能去。由造字功能所編輯的檔案是可產生256個字符或圖案，而每個已設計的字符是可以直接由按鍵取出及顯示。

當然，用者是可以選用系統預置的文字或符號圖案，其中尤以OLD ENGLISH 和KUDZU 字形是極富古典英文字母的書寫藝術。

此外，還有高瘦肥矮的英文字母，以適應不同用戶的需要。通過GRAPHIC WRITER 子程式便可在同一文字檔案內可處理及寫入不同形式的字形與圖案。

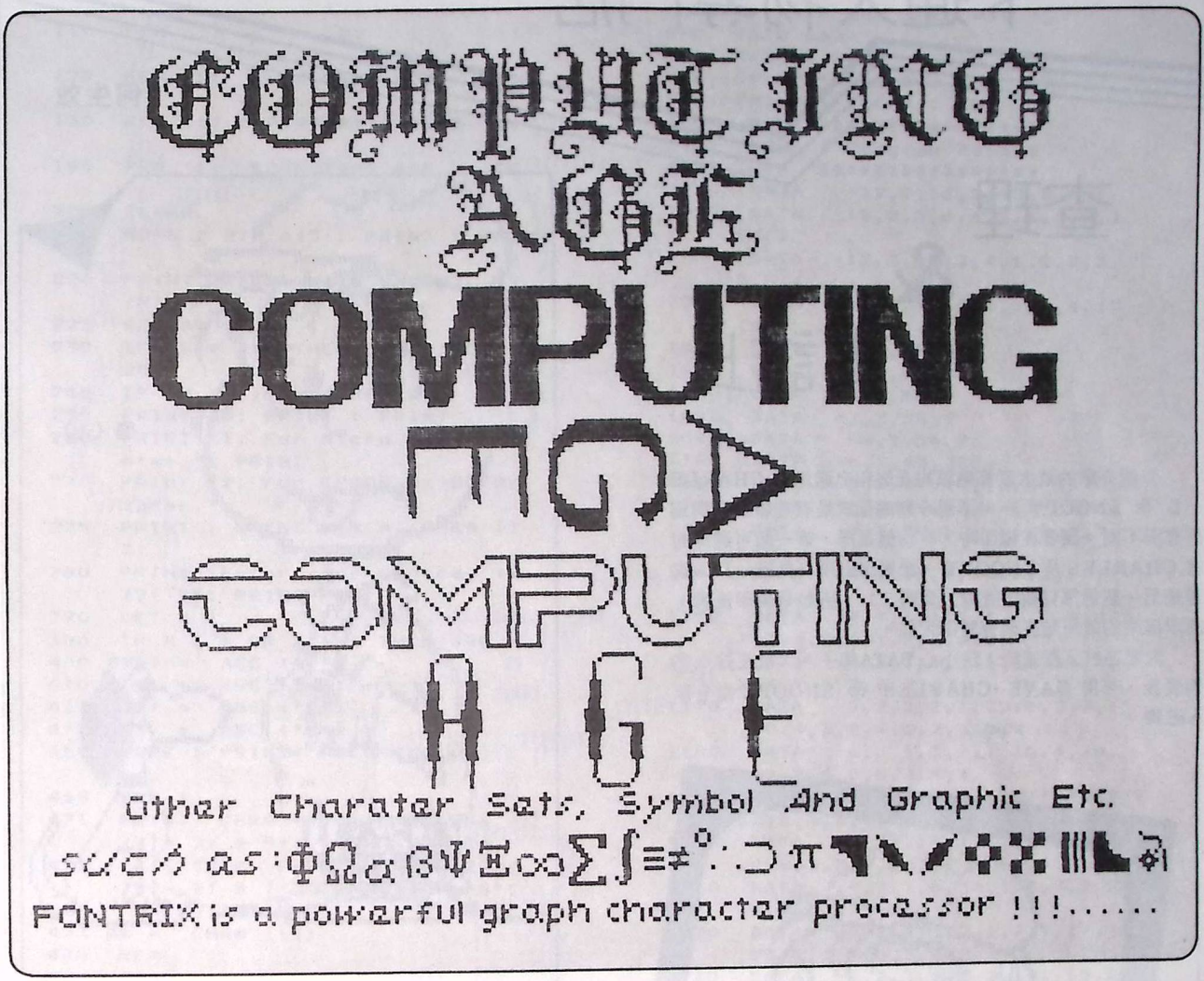
每個字組的矩陣是可由用者置定，而且所有造字功能按鍵是同時清楚列出於螢幕上，好讓用者不須緊記所有造字指令。

GRAPHIC WRITER

FONTRIX 的最特別處，便是利用APPLE 的高解像畫面模擬成一張信紙，好讓用者隨意揮筆！

那麼你可能會懷疑單單一個APPLE 高解像螢幕，是祇可顯示有限的文字圖案！

利用FONTRIX 繪畫的文字及圖畫例子。



因此，GRAPHIC WRITER 是提供GRAFFILE 檔案功能。其方法是由用者先格式一個GRAFFILE 圖案螢幕檔案，而一個高解像畫面是需要（5 × 6）共30 SECTOR 的磁碟容量。

所以，用者是可建立由30至480 SECTOR 的GRAF－FILE 檔案。這也意旨GRAPHIC WRITER 是可以借助磁碟的資料取存媒介，去模擬成超過 4 × 5 個高解像畫面顯示。

正因如此，由一頁高解像螢幕轉至另一頁螢幕時，電腦是會自動從磁碟中寫入和讀出另一頁高解像螢幕的資料內容。這也是GRAPHIC WRITER 被周邊附件拖慢其運行速度之弊端。

還有，GRAPHIC WRITER 也提供了(按CTL－A)足夠的編輯指令提示和強勁的圖案處理功能，其中包括可調校每字間的距離與及字形底色等等。

當然，每個已編輯完成的GRAFFILE 檔案是可通過(P)指令鍵去列印出來。

綜合

基本上，FONTRIX 是一個構思良好的圖案與文字編輯器，但可惜它整個系統大部份是以BASIC 語言所編寫，其程序運行速度上也有所折扣。

此外，系統附送的字形檔案是不足以編輯一篇圖文並茂的文章印本。若用者想添置一些特別的圖案或文字檔案，那麼便要自行設計或付出代價去購買了。因爲在FONTRIX 推出後，美國有很多軟件公司已紛紛替FONTRIX 設計了很多精緻生動的圖像檔案以供FONTRIX 用戶的選購。

還有，早期由美國BORDERBUND 軟件公司推出的PRINT SHOP 程式，其性質與FONTRIX 相似。無論在速度、功能與圖案檔案提供上，前者皆比後者的表現爲佳。那麼爲何這篇文章還介紹FONTRIX 給各位讀者呢！

因爲整個FONTRIX 系統中的GRAPHIC WRITER 是一個多頁高解像螢幕之模擬，其技巧與構思也極爲新意。若FONTRIX 有足夠的圖像檔案所支持，則FONTRIX 必成爲一個高效能圖案與文字的處理系統。■

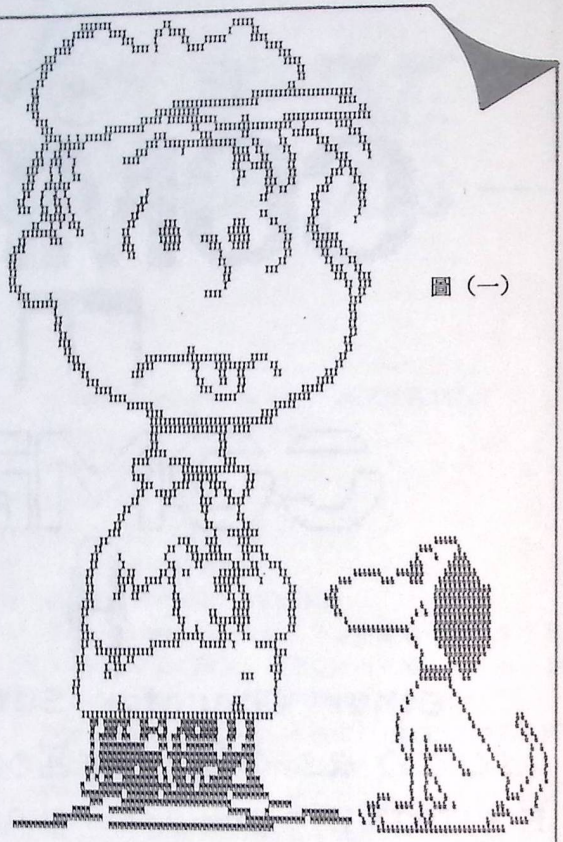
卡通人物專門店

何生效

查理 & 史諾比

介期介紹的是大家都熟識的查理和史諾比（CHARLE—S & SNOOPY）。不過今期的程式是有些特別，與過往有所不同。讀者在編印時，有兩個選擇，第一個可選擇的將 CHARLES 及 SNOOPY 一起編印出來（見圖一），或者是另一個選擇只編印查理（見圖二）。所如是編印後者，則印出來的效果是查理會變得大些。

大家在打入程式時，請小心 DATA句子。大家在打入完列表後，可用 SAVE CHARLES & SNOOPY 指令存入磁碟。



圖（一）

圖（二）

CALENDAR OF 1985

JAN

SU	MO	TU	WE	TH	FR	SA
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

FEB

SU	MO	TU	WE	TH	FR	SA
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28		

MAR

SU	MO	TU	WE	TH	FR	SA
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

APR

SU	MO	TU	WE	TH	FR	SA
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

MAY

SU	MO	TU	WE	TH	FR	SA
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

JUNE

SU	MO	TU	WE	TH	FR	SA
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						

JULY

SU	MO	TU	WE	TH	FR	SA
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

AUG

SU	MO	TU	WE	TH	FR	SA
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

SEP

SU	MO	TU	WE	TH	FR	SA
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

OCT

SU	MO	TU	WE	TH	FR	SA
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

NOV

SU	MO	TU	WE	TH	FR	SA
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

DEC

SU	MO	TU	WE	TH	FR	SA
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

查理與史諾比

```
100  REM  :: CHARLES & SNOOPY  ::
110  REM  ::          BY        ::
120  REM  ::     HO SANG HOU    ::
130  REM  :: COPYRIGHT(C)1985   ::
140  REM  ::  COMPUTING AGE     ::
200  CLEAR
210  HOME : DIM A(3): PRINT : PRINT
220  PRINT "Print with SNOOPY. (Y
     /N) ? ";
222  GET A$
230  IF A$ = "Y" THEN S = 1: GOTO
     250
240  IF A$ <  > "N" THEN 222
250  PRINT A$: PRINT : PRINT
260  PRINT "1. For Micro Line Pri
     nter.": PRINT
270  PRINT "2. For EPSON MX-80 Pr
     inter."
275  PRINT : PRINT :A$ =  CHR$ (1
     3)
280  PRINT "Enter your choice. (1
     /2) ?": PRINT
290  GET X
300  IF X < 1 OR X > 2 THEN 290
400 A(0) =  ASC (" ")
410 A(1) =  ASC ("X")
411 A(2) =  ASC ("M")
412 A(3) =  ASC ("&")
450  HOME : PRINT "PRINTING......
     "
460  PR# 1
471  PRINT  CHR$ (27); CHR$ (56 +
     (X = 2) * 8); CHR$ (27)"E"
472  IF S THEN  PRINT  CHR$ (29 -
     (X = 2) * 14); CHR$ (27)"0";
      CHR$ (27)"F
473 A$ =  CHR$ (13)
480  REM
500  READ X
510  IF X < 1 THEN 580
520  IF C THEN 500
530 Y = (Y + 1) * (Y < 1)
531 Q = 0:Q = (V > 88) * (Y = 1) *
     (D = 0) + (D = 1) * (Y = 1) *
     2
535  FOR Z = 1 TO X
540  PRINT  CHR$ (A(Y + Q));: NEXT
550  GOTO 500
560  PRINT A$; TAB(  ABS (X) + S *
     12)
565 C = 0:D = 0:Y = 0:V = V + 1
570  GOTO 500
580  IF X =  - 999 THEN 800
590  IF X < 0 THEN 560
600 C = 1 - S:D = 1
610  GOTO 500
800  FOR X = 1 TO 5
810  PRINT A$;
820  NEXT X
830  PR# 0: PRINT
840  PRINT "Do you want another o
     ne. (y/n) ?"
850  PRINT : PRINT
860  GET A$
870  IF A$ = "Y" THEN  RUN
880  IF A$ = "N" THEN 910
890  GOTO 860
900  RESUME 800
910  PRINT "Good Bye  !! !"
920  PRINT : CLEAR : END
1000  REM  ****************
1001  REM  * Program data *
1002  REM  ****************
1010  DATA   -17,7,11,2,13,5
1020  DATA  -15,5,3,4,6,2,1,4,9,1
     ,4,2
1030  DATA  -13,3,10,3,4,1,5,2,3,
     5,5,4
1040  DATA  -11,4,13,4,1,1,6,4,13
     ,4
1050  DATA  -9,3,48,3
1060  DATA  -8,3,50,4
1070  DATA  -8,2,54,2
1080  DATA  -7,2,55,2
1090  DATA   -6,2,54,2
1100  DATA  -6,2,46,22
1110  DATA  -5,3,29,19,17,4
1120  DATA  -6,2,25,10,34,2
1130  DATA  -7,3,20,5,1,11,5,2,7,
     19
1140  DATA  -8,3,11,9,10,2,3,17,2
     ,1,6,4
1150  DATA  -9,4,1,1,1,1,1,6,9,3,
     7,3,6,2,6,1,2,6,6,3
1160  DATA  -8,2,2,6,13,3,1,16,3,
     2,4,2,6,3,6,2
1170  DATA  -7,2,2,2,1,1,14,3,7,1
     ,10,4,1,2,4,2,7,3,5,2
1180  DATA  -6,3,1,3,2,1,10,4,20,
     4,2,2,4,2,8,3,4,1
1190  DATA  -5,3,2,1,1,1,2,1,34,2
     ,2,5,3,2,9,3
1200  DATA  -5,2,3,1,1,1,2,2,34,2
     ,4,2,3,2,7,3,1,2
1210  DATA  -4,3,2,6,36,1,5,3,2,2
     ,6,2,4,2
1220  DATA  -4,2,2,1,2,1,1,1,17,1
     ,25,2,1,2,6,3,6,2
1230  DATA  -3,2,3,1,2,2,1,1,15,2
     ,24,2,9,2,8,2
1240  DATA  -3,3,2,2,1,2,1,1,14,2
     ,36,2,8,2
1250  DATA  -8,6,1,1,12,2,26,2,9,
     2,8,1
1260  DATA   -6,2,2,1,1,1,1,3,10,
     2,6,3,12,3,5,2,6,2,8,1
1270  DATA  -4,8,4,2,8,2,6,5,10,5
     ,6,2,4,3
1280  DATA  -3,2,5,3,13,2,6,5,4,4
     ,2,5,7,2,4,6
1290  DATA  -2,3,21,3,6,4,6,3,2,4
     ,7,2,5,6
1300  DATA  -1,2,24,2,18,1,13,2,6
     ,2,2,2
1310  DATA  -1,2,44,2,11,2,8,2,3,
     2
1320  DATA  -1,2,45,2,7,3,11,3,2,
     2
1330  DATA  -1,2,45,2,22,3,2,3
1340  DATA  -1,3,44,1,25,2,1,3
1350  DATA  -3,3,4,1,33,4,27,1,1,
     2
1360  DATA  -4,8,63,3
1370  DATA  -11,1,63,2
1380  DATA  -11,2,62,2
```

查理與史諾比

```
1390  DATA  -11,2,62,2
1400  DATA  -12,2,61,2
1410  DATA  -12,3,59,2
1420  DATA  -13,2,25,2,32,2
1430  DATA  -14,3,14,19,4,4,15,2
1440  DATA  -15,3,11,4,8,2,3,9,3,
      2,12,2
1450  DATA  -17,2,9,3,11,1,4,2,2,
      2,6,2,9,3
1460  DATA  -18,3,21,2,3,2,2,2,15
      ,4
1470  DATA   -20,4,19,3,5,2,12,5
1480  DATA  -23,4,19,5,11,5
1490  DATA  -27,6,26,5
1500  DATA  -31,4,21,4
1510  DATA  -32,27
1520  DATA  -32,17,1,4
1530  DATA  -32,2,13,2
1540  DATA  -32,2,13,2
1550  DATA  -31,5,11,2
1560  DATA  -28,4,4,5,7,3
1570  DATA  -28,1,10,15
1580  DATA  -28,3,8,8,6,2
1590  DATA  -30,3,4,4,2,1,1,4,4,2

1600  DATA  -29,2,2,5,5,1,2,1,1,2
      ,2,2
1610  DATA  -28,2,5,2,6,3,3,4
1620  DATA  -26,2,15,2,6,1
1630  DATA  -25,2,17,1,6,2
1640  DATA  -24,2,18,2,6,2
1650  DATA  -23,3,20,1,2,7
1660  DATA   -21,2,15,3,3,3,4,3
1665  DATA  0,39,5,0
1670  DATA  -20,2,14,2,2,3,2,3,4,
      5
1675  DATA  0,33,3,5,1,0
1680  DATA  -19,2,4,1,9,1,5,9,2,2
      ,2,2
1685  DATA  0,31,1,3,1,3,3,0
1690  DATA  -19,2,5,1,9,2,4,2,6,2
      ,2,2,2,3
1695  DATA  0,27,3,5,5,0
1700  DATA  -18,2,5,1,1,2,5,2,1,1
      ,1,2,6,9,4,2
1705  DATA  0,14,6,3,6,4,7,0
1710  DATA  -18,2,5,4,2,4,1,2,8,6
      ,4,1,4,2
1715  DATA  0,10,3,6,3,3,3,3,9,0
1720  DATA  -18,2,5,2,1,4,5,1,8,4
      ,1,2,9,2
1725  DATA  0,7,2,20,12,0
1730  DATA  -18,2,4,2,2,2,7,2,10,
      2,1,2,8,2
1735  DATA  0,6,3,20,12,0
1740  DATA  -17,3,17,2,11,4,8,2
1745  DATA  0,7,2,17,1,2,12,0
1750  DATA  -17,2,18,2,6,2,3,4,8,
      2
1755  DATA  0,8,2,15,1,1,1,1,12,0

1760  DATA  -17,1,20,2,5,9,8,2
1765  DATA   0,10,1,13,1,4,12,0
1770  DATA  -17,2,20,1,2,1,4,2,2,
      2,9,2
1775  DATA  0,11,13,5,12,0
1780  DATA  -17,3,17,7,3,3,2,2,5,
      3
1785  DATA  0,26,1,4,12,0
1790  DATA  -17,2,1,2,10,7,4,6,1,
      1,4,5
1795  DATA  0,29,1,4,11,0
1800  DATA  -17,1,2,13,26,3
1805  DATA  0,28,1,4,9,0
1810  DATA  -16,2,5,3,1,2,30,2
1815  DATA  0,29,1,5,7,0
1820  DATA  -16,2,4,1,1,3,31,3
1825  DATA  0,29,7,1,4,0
1830  DATA  -16,1,5,2,28,3,4,2
1835  DATA  0,29,7,2,2,0
1840  DATA  -15,2,42,1
1845  DATA  0,29,1,6,1,0
1850  DATA  -15,2,42,1
1855  DATA  0,28,1,8,1,0
1860  DATA  -16,1,8,1,24,2,7,1
1865  DATA  0,27,1,10,1,0
1870  DATA  -16,1,8,2,23,3,6,1
1875  DATA  0,26,1,12,1,0
1880  DATA  -17,42
1885  DATA  0,26,1,14,1,9,2,0
1890  DATA  -19,2,1,6,2,2,2,2,3,8
      ,2,2,2,2
1895  DATA  0,30,1,15,1,8,1,1,1,0

1900  DATA  -19,1,2,3,1,1,3,4,1,1
      ,2,4,1,4,2,2,2,2
1905  DATA  0,30,1,2,1,13,1,8,1,1
      ,1,0
1910  DATA  -19,1,1,4,1,2,2,2,2,2
      ,1,2,1,7,2,2,2,2
1915  DATA  0,30,1,2,1,2,1,11,1,8
      ,1,1,1,0
1920  DATA  -19,1,1,10,3,2,1,2,1,
      6,3,1,2,3
1925  DATA  0,31,1,1,1,2,1,1,4,7,
      1,8,1,1,1,0
1930  DATA  -19,1,1,14,1,11,1,7
1935  DATA  0,32,2,2,2,4,1,7,1,7,
      1,2,1,0
1940  DATA  -19,1,1,14,2,2,1,12,1
      ,2
1945  DATA  0,33,1,2,1,6,1,7,1,6,
      1,2,1,0
1950  DATA  -18,2,1,8,1,4,1,1,1,7
      ,3,4,2,2
1955  DATA  0,33,1,2,2,6,2,6,1,5,
      1,2,1,0
1960  DATA  -18,1,1,1,1,2,2,8,1,1
      ,1,3,1,1,1,1,5,3,3
1965  DATA  0,32,1,2,1,1,1,14,1,3
      ,1,3,1,0
1970  DATA  -17,1,3,16,1,4,1,14
1975  DATA  0,31,1,2,1,2,1,13,1,2
      ,1,3,1,0
1980  DATA  -17,9,1,7,1,14,1,8
1985  DATA  0,23,3,4,1,2,2,2,1,12
      ,3,3,1,0
1990  DATA  -19,3,4,24,8,5
1995  DATA  0,16,3,3,4,1,1,2,3,1,
      1,8,2,6,1,0
2000  DATA  -15,4,4,8,18,5,3,1,7,
      8
2005  DATA  0,4,1,1,2,9,1,1,1,12,
      4,4,1,0
2010  DATA  -11,11,31,13,1,9
2015  DATA  0,2,1,1,1,1,1,6,1,1,2
      ,11,1,4,4,0
2020  DATA  -2,12,9,36
2025  DATA  0,22,27
2030  DATA  -999
2040  END
```

掀開蘋果高解像畫面的神秘面紗

圖形繪畫法研究（四）

麥法基專欄

當我們能夠把各種圖形繪畫出來，這便是動畫設計的基本第一步。第二步却是動畫的主要部份：怎樣令一個或以上的圖形在畫面上移動？

當然我們已經有一個遊戲的概念，這個概念包括物體移動方向、快慢、何時消失、何時重現、測試碰撞、移動路綫等。一般來說，這些程序需要程式去處理。不過以上所指的都是一些不斷變更的資料，倘若我們把資料初值輸入，然後交由一個指定程式去負責，則我們只要集中在遊戲其他方面便可以了。動畫程式（ANIMATION ROUTINE）和動畫編輯（ANIMATION EDITOR）便是用以處理這些複雜的資料。

記憶位址由$8F00至$95FF之間是所有動畫程式所在，而其中由$9000至$9200之間是整個程式心臟，在未討論ANIMATION EDITOR之前，首先讓我們了解一下$9000至$9200之間的ANIMATION ROUTINE。

這裏要指出動畫程式並非十全十美，由於要照顧大部份應用，一些特殊的需要例如PAGE FLIPPING，在彩色背景繪畫彩色圖形都無法進行。不過如果深入認識動畫程式，對這些要求都可以作出適當改變加以應用。

圖一是動畫程式一個大概運作過程，而刊出的LISTING已經加上註解，讀者可以從中參考每個部份。

首先談談第一部份：

OBJECT LIST的測試

從OBJECT LIST測試物體是否需要移動

OBJECT LIST由$9400至$9420共存有32個物體的圖形號碼。但並非只有號碼而已。每一個位元組顯示圖形號碼值，而動畫程式根據這些號碼去順序找尋圖形表。但並非需要處理32個物體，倘若我們只有一個物體的話，則第二個位元組將會被存入$FF數值，表示所有動畫操作完畢。倘若在動畫過程當中把任何一個位元組值由圖形號碼改爲大於$7F而少於$FF，則動畫程式會停止該物體移動，但不加以消圖。

這些工作程序可以由第40至50行觀察出來，程式由第一個物體開始工作，所以40至42行都把SHAPCOUNT等暫存入零數值（電腦數值以零作爲開始），第44行從OBJECT LIST內提取數值，若少於128（$80）表示有物體需要移動，程式會跳到ANIMSTART進行。第45行如果該數值大於128却等於255的話，則跳到END部份完畢。如果兩者皆不是，則當然需要停止該物體移動，停止的意思是指圖形仍然保留畫面上不加以消去，只不過不移動吧了。這個動畫程式以XDRAW爲主，換言之物體圖形必須預先存在畫面上才作消圖，繪畫新圖形工作。所以第49行程式會直接跳去NEXT部份進行第二個物體移動程序，而先前一個物體圖形便不加以消圖了。

由此可見如果我們希望移動的物體在某些情況下靜止下來，只要在相對的OBJECT LIST內存入$80（128）數值便成。重新移動物體則在同一位置存回圖形號碼。

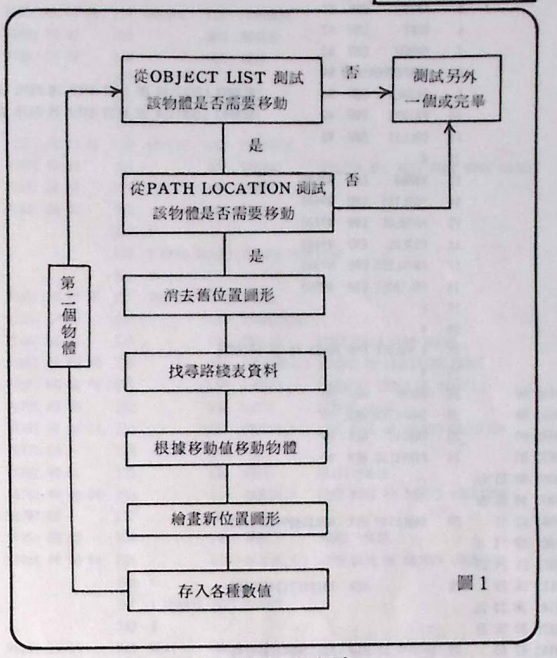


圖1

PATH LOCATION的測試

從PATH LOCATION測試該物體是否需要移動

由OBJECT LIST去控制物體靜止或移動並非萬全之策。有時我們基於某些原因不能改變該LIST的值，則可以在測試路綫當中進行。第53至59行便是這些工作程序所在。PATH LOCATION由$9440至$949F，每個物體佔用3個位元組。第一個存入路綫號碼，第二個存入路綫表低位元，而第三個存入路綫表高位元。PATH LOCATION實際上是路綫表一個指針，它隨著物體移動指向路綫表移動值位址，讓動畫程式知道如何跟隨路綫表移動，就像火車頭一樣帶動整列火車開行。我們可以擁有3條路綫表順序跟隨，這是第一個位元組的工作，它表示現在用那一條路綫。不過倘若我們把它的值改爲$80（128），則物體一樣會靜止下來。

第52行把圖形號碼存入SHNUM以供繪圖之用（參考第廿三期），然後把INDEXTMP取出相對的索引值，第56行測試PATH LOCATION個別物體的第一個位元組值是否大過127，若是則該物體同樣靜止下來。

爲甚麼要用INDEXTMP呢？這是一個表，在上一部份的OBJECT LIST每個物體只佔用一個位元組用來存入圖形號碼，而這裏PATH LOCATION却用3個位元組去存放數值。換言之，Y－REGISTER無法同時期作爲兩部份索引之用，所以要另外設立一個INDEXTMP索引表以供運用。假如我們觀察27行的INDEXTMP，它的數值是3、6、9如此類推。這樣當SHAPCOUNT是零時，則INDEX－TMP的值是3，這時再把3存入INDEX 1內，便可以找到PATH LOCATION的索引值了。

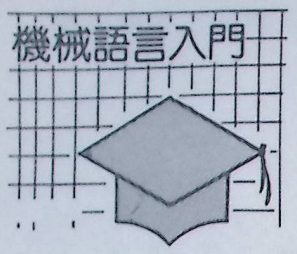
動畫程式比較複雜，下一期再討論其他部份。■

圖形繪畫法研究（四）

```
                1           ORG  $9000
                2  *
                3  *
                4  XBYTE    EQU  $0
                5  XBIT     EQU  $1
                6  VERT     EQU  $2
                7  SHNUM    EQU  $3
                8  SHAPECOUNTEQU $4
                9  PATHLO   EQU  $6         ;MEMORY LOCATION OF NEXT STEP IN PATH LO
               10  PATHHI   EQU  $7         ;MEMORY LOCATION OF NEXT STEP IN PATH HI
               11  COLLIS   EQU  $B
               12  *
               13  XDRAW    EQU  $8F00
               14  OBJLIST  EQU  $9400
               15  PATHLOC  EQU  $9440
               16  OBJLOC   EQU  $94A0
               17  PATHLIST EQU  $9500
               18  COLTABLE EQU  $95E0
               19  *
               20  *
               21  * VALUES FOR INDEX TO 32 OBJECTS
               22  *
9000: 00       23  OBINF    HEX  00
9001: 00       24  SHAPETEMPHEX  00
9002: 00       25  INDEX1   HEX  00
9003: 03       26  PTHVALUE HEX  03
9004: 00 03 06
9007: 09 0C 0F
900A: 12 15    27  INDEXTMP HEX  000306090C0F1215
900C: 18 1B 1E
900F: 21 24 27
9012: 2A 2D    28           HEX  181B1E2124272A2D
9014: 30 33 36
9017: 39 3C 3F
901A: 42 45    29           HEX  303336393C3F4245
901C: 48 4B 4E
901F: 51 54 57
9022: 5A 5D    30           HEX  484B4E5154575A5D
               31  *
               32  * TABLE FOR MOVEING OBECTS 2 UNITS
               33  *
9024: 00 00 00
9027: 00 00 01
902A: 01       34  R2BYTE   HEX  00000000000101
902B: 02 03 04
902E: 05 06 00
9031: 01       35  R2BIT    HEX  02030405060001
9032: FF FF 00
9035: 00 00 00
9038: 00       36  L2BYTE   HEX  FFFF0000000000
9039: 05 06 00
903C: 01 02 03
903F: 04       37  L2BIT    HEX  05060001020304
               38  *
               39  *
9040: A9 00    40  ANIMATE  LDA  @$0
9042: 85 04    41           STA  SHAPECOUNT ;COUNTING FOR OBJECTS
9044: 8D 01 90 42           STA  SHAPETEMP
9047: A4 04    43  ANIMATE1 LDY  SHAPECOUNT
9049: B9 00 94 44           LDA  OBJLIST,Y  ;TAKE VALUE OF OBJECT LIST
904C: 10 08    45           BPL  ANIMSTART  ;IF <128 THEN CONTINUE UPDATE
904E: C9 FF    46           CMP  @$FF       ;IF @$FF THEN END OF ANIMATION UPDATE
9050: F0 03    47           BEQ  END
9052: 4C 83 91 48           JMP  NEXT       ;IF >127 THEN OBJECT INACTIVE
9055: 60       49  END      RTS             ;END OF ANIMATION
               50  *
               51  *
9056: 85 03    52  ANIMSTARTSTA  SHNUM      ;STORE VALUE OF SHAPE NUMBER
9058: B9 04 90 53           LDA  INDEXTMP,Y  ;TEMP VARIABLE TO NEED INDEX
905B: 8D 02 90 54           STA  INDEX1
905E: AC 02 90 55           LDY  INDEX1
9061: B9 40 94 56           LDA  PATHLOC,Y  ;TAKE VALUE OF PATH LOCATION'S 1ST BYTE
9064: 10 03    57           BPL  ANIM2      ;IF <128 THEN CONTINUE UPDATE
9066: 4C 83 91 58           JMP  NEXT       ;THIS OBJECT INACTIVE, DO NEXT ONE
               59  *
               60  * START ANIMATION UPDATE *
               61  *
9069: B9 A0 94 62  ANIM2    LDA  OBJLOC,Y   ;1ST BYTE OF OBJECT LOCATION
906C: 85 00    63           STA  XBYTE      ;X-BYTE POSITION
906E: C8       64           INY             ;NEXT BYTE
906F: B9 A0 94 65           LDA  OBJLOC,Y   ;VALUE
9072: 85 01    66           STA  XBIT       ;X-BIT POSITION
9074: C8       67           INY             ;NEXT BYTE
9075: B9 A0 94 68           LDA  OBJLOC,Y   ;VALUE
9078: 85 02    69           STA  VERT       ;Y CO-ORDINATE VALUE
907A: 20 00 8F 70  ERASE    JSR  XDRAW      ;ERASE OBJECT FIRST
               71  *
               72  *
               73  *  FIND PATH'S VALUE FOR OBJECT MOVEMENT
               74  *
907D: AE 02 90 75  FINDPATH LDX  INDEX1
9080: E8       76           INX
9081: BD 40 94 77           LDA  PATHLOC,X  ;NEXT PATH LOCATION BYTE
9084: 85 06    78           STA  PATHLO     ;STORE TO LO POINTER OF PATH'S ADDRESS
9086: E8       79           INX
9087: BD 40 94 80           LDA  PATHLOC,X
908A: 85 07    81           STA  PATHHI     ;STORE TO HI POINTER OF PATH ADDRESS
908C: CA       82           DEX
908D: FE 40 94 83           INC  PATHLOC,X  ;TEST IF PATH'S MEMORY REACH BOUNDRY?
9090: D0 04    84           BNE  CONTIN     ;NO, CONTINUE UPDATE
9092: E8       85           INX             ;YES,
9093: FE 40 94 86           INC  PATHLOC,X  ;INCREASE HI POINTER BY ONE
               87  *
               88  * CHECK IF WHOLE PATH FINISH?
               89  *
9096: A0 00    90  CONTIN   LDY  @$00
9098: B1 06    91           LDA  (PATHLO),Y ;PATH'S VALUE=0?(END OF THIS PATH)
909A: D0 52    92           BNE  MOVE       ;NO, TO OBJECT MOVEMENT
909C: AE 02 90 93           LDX  INDEX1     ;YES, TAKE PRESENT INDEX VALUE
909F: FE 40 94 94           INC  PATHLOC,X  ;INCREASE TO NEXT PATH
90A2: BD 40 94 95           LDA  PATHLOC,X
               96  *
               97  *
               98  * NOW WHOLE PATH FINISHED, CHECK IF ANOTHER PATH NEEDED
               99  *
90A5: 0A      100  CHECKPATHASL            ;TIMES 2
90A6: 18      101          CLC
90A7: 6D 02 90 102          ADC  INDEX1
90AA: 6D 02 90 103          ADC  INDEX1
90AD: 65 04    104          ADC  SHAPECOUNT ;ADD INDEX1 2 TIMES AND SHAPECOUNT
90AF: A8       105          TAY             ;TO OBTAIN Y INDEX FOR PATH LIST
90B0: B9 00 95 106          LDA  PATHLIST,Y ;STORE VALUE
90B3: 85 07    107          STA  PATHHI     ;TO HI POINTER OF PATH ADDRESS FIRST
90B5: 30 14    108          BMI  CHECKFLAG  ;VALUE > @$80, BYTE IS A FLAG
               109 *
               110 *
               111 * BYTE IN THE PATH LIST IS NOT A FLAG, SO NEXT PATH NEEDED
               112 *
90B7: C8      113  NEXTPATH INY            ;NEXT PATH
90B8: B9 00 95 114          LDA  PATHLIST,Y ;TAKE LO POINTER VALUE OF PATHLIST
90BB: AC 02 90 115          LDY  INDEX1     ;RESET INDEX
90BE: C8       116          INY
90BF: 99 40 94 117          STA  PATHLOC,Y  ;STORE VALUE TO 2ND PATH LOCATION BYTE
90C2: C8       118          INY
90C3: A5 07    119          LDA  PATHHI     ;TAKE HI POINTER VALUE OF PATHLIST
90C5: 99 40 94 120          STA  PATHLOC,Y  ;STORE TO 3RD PATH LOCTION BYTE
90C8: 4C 7D 90 121          JMP  FINDPATH   ;GO BACK TO FIND PATH VALUE FOR MOVEMENT
               122 *
               123 * CHECK FLAG VALUE TO DETERMINE WHAT TO DO NEXT
               124 *
90CB: 24 07    125  CHECKFLAGBIT  PATHHI    ;VALUE <@$8F?
90CD: 50 E8    126          BVC  NEXTPATH   ;YES, NEXT PATH NEEDED
90CF: C9 FF    127          CMP  @$FF       ;=@$FF?
90D1: F0 10    128          BEQ  CHECKFLAG2 ;YES, REPEAT SAME PATH
90D3: AC 02 90 129          LDY  INDEX1     ;NO, POINT TO 1ST BYTE OF PATH LOCATION
90D6: 99 40 94 130          STA  PATHLOC,Y  ;STORE VALUE WHICH IS >127 TO BYTE
90D9: C9 FE    131          CMP  @$FE       ;VALUE = @$FE?
90DB: F0 03    132          BEQ  NEXT1      ;YES, END DRAW WITHOUT ERASING
90DD: 4C 83 91 133          JMP  NEXT       ;END DRAW WITH ALREADY ERASING,TO NEXT ONE
```

圖形繪畫法研究（四）

```
90E0: 4C 65 91  134 NEXT1   JMP  DRAWOB    ;THEREFORE DRAW OBJECT
                135 *
                136 *
                137 * REPEAT SAME PATH
                138 *
90E3: AE 02 90  139 CHECKFLAG2LDX INDEX1    ;RESET INDEX
90E6: A9 00     140         LDA  #$00
90E8: 9D 40 94  141         STA  PATHLOC,X  ;PATH NO.1 REPEAT
90EB: 4C A5 90  142         JMP  CHECKPATH  ;PUT IN PATH PARAME"ERS
                143 *
                144 *
                145 * MOVE OBJECT ACCORDING TO PATH VALUE
                146 *
90EE: C8        147 MOVE    INY
90EF: 8C 01 90  148         STY  SHAPETEMP
90F2: 8D 03 90  149         STA  PTHVALUE   ;PRESENT PATH VALUE
90F5: 4E 03 90  150         LSR  PTHVALUE   ;SHIFT RIGHT TO TEXT BIT
90F8: 90 0F     151         BCC  MOVER2     ;IF ZERO THAN SHOULD BE 2 UNITS' RIGHT
                152 *
                153 * MOVE 1 UNIT RIGHT
                154 *
90FA: A4 01     155 MOVER1  LDY  XBIT       ;XBIT VALUE
90FC: B9 88 91  156         LDA  R1BYTE,Y   ;FIND FROM MOVE TABLE RIGHT 1 BYTE VALUE
90FF: 18        157         CLC
9100: 65 00     158         ADC  XBYTE      ;ADD TO XBY"E VALUE
9102: 85 00     159         STA  XBYTE      ;STORE BACK TO XBYTE VALUE
9104: B9 8F 91  160         LDA  R1BIT,Y    ;FIND FROM MOVE TBLE RIGHT 1 BIT VALUE
9107: 85 01     161         STA  XBIT       ;STORE TO XBIT
                162 *
                163 * MOVE 2 UNITS RIGHT
                164 *
9109: 4E 03 90  165 MOVER2  LSR  PTHVALUE   ;SHIFT PATH VALUE AGAIN
910C: 90 0F     166         BCC  MOVEL1     ;IF ZERO THEN TEST MOVE LEFT
910E: A4 01     167         LDY  XBIT       ;NO, TAKE PRESENT XBIT VALUE
9110: B9 24 90  168         LDA  R2BYTE,Y   ;TAKE VALUE FROM LEFT BYTE TABLE
9113: 18        169         CLC
9114: 65 00     170         ADC  XBYTE
9116: 85 00     171         STA  XBYTE
9118: B9 2B 90  172         LDA  R2BIT,Y
911B: 85 01     173         STA  XBIT
                174 *
                175 *MOVE 1 UNIT LEFT
                176 *
911D: 4E 03 90  177 MOVEL1  LSR  PTHVALUE
9120: 90 0F     178         BCC  MOVEL2
9122: A4 01     179         LDY  XBIT
9124: B9 96 91  180         LDA  L1BYTE,Y
9127: 18        181         CLC
9128: 65 00     182         ADC  XBYTE
912A: 85 00     183         STA  XBYTE
912C: B9 9D 91  184         LDA  L1BIT,Y
912F: 85 01     185         STA  XBIT
                186 *
                187 * MOVE 2 UNITS LEFT
                188 *
9131: 4E 03 90  189 MOVEL2  LSR  PTHVALUE
9134: 90 0F     190         BCC  MOVED1     ;MOVE 1 UNIT DOWN
9136: A4 01     191         LDY  XBIT
9138: B9 32 90  192         LDA  L2BYTE,Y
913B: 18        193         CLC
913C: 65 00     194         ADC  XBYTE
913E: 85 00     195         STA  XBYTE
9140: B9 39 90  196         LDA  L2BIT,Y
9143: 85 01     197         STA  XBIT
                198 *
                199 * MOVE 1 UNT DOWN
                200 *
9145: 4E 03 90  201 MOVED1  LSR  PTHVALUE
9148: 90 02     202         BCC  MOVED2
914A: E6 02     203         INC  VERT
                204 *
                205 * MOVE 2 UNIT DOWN
                206 *
914C: AD 03 90  207 MOVED2  LDA  PTHVALUE
914F: 90 04     208         BCC  MOVEU1     ;MOVE 1 UNIT UP
9151: E6 02     209         INC  VERT
9153: E6 02     210         INC  VERT
                211 *
                212 * MOVE 1 UNIT UP
                213 *
9155: 4E 03 90  214 MOVEU1  LSR  PTHVALUE
9158: 90 02     215         BCC  MOVEU2
915A: C6 02     216         DEC  VERT
                217 *
                218 * MOVE 2 UNITS UP
                219 *
915C: 4E 03 90  220 MOVEU2  LSR  PTHVALUE
915F: 90 04     221         BCC  DRAWOB     ;FINISH ALL TEST THEN DRAW OBJECT
9161: C6 02     222         DEC  VERT
9163: C6 02     223         DEC  VERT
                224 *
                225 * DRAW OBJECT IN NEW POSITION
                226 *
9165: 20 00 8F  227 DRAWOB  JSR  XDRAW
9168: A4 04     228         LDY  SHAPECOUNT
916A: A5 0B     229         LDA  COLLIS     ;COLLISION FLAG VALUE
916C: 99 E0 95  230         STA  COLTABLE,Y ;STORE TO COLLISION TABLE
916F: AC 02 90  231         LDY  INDEX1     ;PRESENT INDEX OF OBJECT
9172: A5 00     232         LDA  XBYTE      ;XBYTE VALUE
9174: 99 A0 94  233         STA  OBJLOC,Y   ;1ST BYTE OF OBJECT LOCATION
9177: C8        234         INY
9178: A5 01     235         LDA  XBIT       ;XBIT VALUE
917A: 99 A0 94  236         STA  OBJLOC,Y   ;2ND BYTE OF OBJECT LOCATION
917D: C8        237         INY
917E: A5 02     238         LDA  VERT       ;VERT VALUE
9180: 99 A0 94  239         STA  OBJLOC,Y   ;3RD BYTE OF OBJECT LOCATION
                240 *
                241 * UPDATE NEXT OBJECT
                242 *
9183: E6 04     243 NEXT    INC  SHAPECOUNT ;NEXT SHAPE NUMBER
9185: 4C 47 90  244         JMP  ANIMATE1   ;JUMP BACK "O UPDATE
                245 *
9188: 00 00 00
918B: 00 00 00
918E: 01        246 R1BYTE  HEX  00000000000001
918F: 01 02 03
9192: 04 05 06
9195: 00        247 R1BIT   HEX  01020304050600
9196: FF 00 00
9199: 00 00 00
919C: 00        248 L1BYTE  HEX  FF000000000000
919D: 06 00 01
91A0: 02 03 04
91A3: 05        249 L1BIT   HEX  06000102030405
                250 *
                251 *
91A4: A9 4C     252 INITDRAW LDA  #$4C      ;CHANGE INSTRUCTION TO JMP
91A6: 8D 7A 90  253         STA  ERASE
91A9: A9 65     254         LDA  #$65
91AB: 8D 7B 90  255         STA  ERASE+1
91AE: A9 91     256         LDA  #$91
91B0: 8D 7C 90  257         STA  ERASE+2    ;THE ABOVE 2 CHANGE PROGRAM TO DRAWOB
91B3: 20 40 90  258         JSR  ANIMATE    ;INITIAL DRAW ALL OBJECTS
91B6: A9 20     259         LDA  #$20
91B8: 8D 7A 90  260         STA  ERASE
91BB: A9 00     261         LDA  #$00
91BD: 8D 7B 90  262         STA  ERASE+1
91C0: A9 8F     263         LDA  #$8F
91C2: 8D 7C 90  264         STA  ERASE+2    ;RESET BACK ALL INSTRUCTIONS
91C5: 60        265         RTS
                266 *
                267 *


--End assembly--

454 bytes

Errors: 0
```



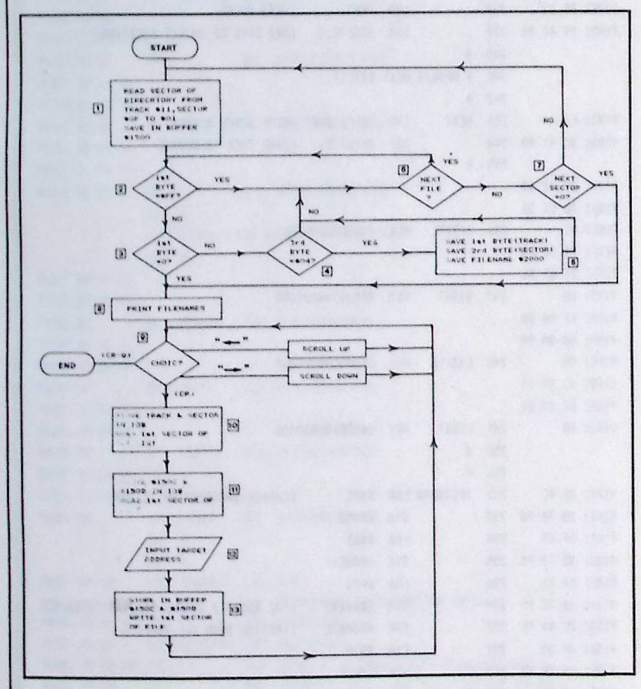
起始地址搬移器

李兆基

用機械語言（MC）編寫程式浪費很多時間，同時偵錯困難，所以大多數程式設計員都採用組合語言（AL）編寫，然後再用軟件編譯成MC APPLE用家對使用LISA 2・5一定不會陌生，這軟件除編譯速度快，還提供額外的語法，編輯容易等優點。筆者經多次使用後發覺它唯一的缺點是目的碼（OBJ・CODE）存入磁碟時起始地址受到一定限制。

原因很簡單，LISA 2・5 將記憶區劃分不同用途，目的碼只佔用＄800至＄17FF。假如程式起始位置加上長度超過＄1800，目的碼就必須放在＄800至＄17FF之間，而最方便就是由＄800開始。例如

ORG ＄4000
OBJ ＄800



說明程式原本（ORGINAL）是由＄4000開始，因此編譯成MC時必須針對此起始地址，但是目的碼（OBJECT CO－DE）却存放在＄800開始。

編譯完成後，將原始碼和目的碼存入磁碟，試執行目的碼，會發現有ERROR，因爲MC存放在FILE的第一個SE－CTOR的第0和第一個BYTE，要程式執行必須改變爲＄00、＄40，或者在執行時打入BRUN FILENAME，A＄4000。如果經常使用的程式便需緊記開始位置或者用SECT－OR ZAP找出存放地址的BYTES，改寫後寫回原來的SECTOR。

筆者針對上述情況特別編寫程式MACHINE CODES OFFSETTER方便各位常用LISA 2・5的讀者。程式執行後會提示放入有MC的磁碟，任意按鍵便會將所有"B"TYPE的FILE列出，如果沒有會重新提示。"←"、"→"和<CR>鍵選擇FILE，螢幕下方顯示FILENAME、ADDRESS和LENGTH，最下方是目的地址，只要鍵入所需ADDRESS並按<CR>就會完成改動，讀者可重覆上述步驟，直至按CONTROL－Q跳出程式爲止。

程式工作原理很簡單，筆者在此略加說明。請參看流程圖：

(1) 開始時跳至RWTS副程式讀入DIRECTORY並存入＄1500至＄15FF記憶區。
(2) 比較每一個FILE DIRECTORY的第零BYTE，是＄FF表示這個FILE已被DELETE，跳至⑥
(3) 如果是＄00表示沒有其他FILE，跳至⑧
(4) 比較第2個BYTE，不等於＄04（"B" TYPE）則跳至⑥
(5) 將第0TRACK和第1 SECTOR分別存入＄1300和＄1400位置開始並存入FILENAME在＄2000地址開始。
(6) 如果有FILE跳回②
(7) 如果SECTOR不等於零，跳回①讀下一個SECTOR。
(8) 印出所有FILENAME。
(9) "←"、"→"令螢幕上下捲動，CTL－Q跳出程式，RETURN選擇FILE。
(10) 把TRACK和SECTOR放入IOB讀取檔名的TRACK／SECTOR LIST。
(11) ＄0C和＄0D是存放FILE的第一個SECTOR的地址，因此把它們放入IOB並讀取。
(12) 要求用家輸入目的地址。
(13) 將目的地址放回＄0C和＄0D取代舊地址，並寫入同一個SECTOR回磁碟內。跳回⑨重覆選擇。

程式鍵入

本程式是用LISA 2・5編寫，懂得ASSEMBLER的用家可以逐行鍵入，然後組合出一個OBJECT CODE檔案。不懂得ASSEMBLER的讀者可按左邊機械碼部份輸入。輸入後，可用下列指令：

BSAVE MACHINE CODES OFFSETTER，A＄801，L＄361存入磁碟。

不想鍵入程式而又想使用本程式的讀者可購買一張同期一齊出版的程式磁碟，內裏收錄有相同名字的FILE。

起始地址搬移器

```
]
!ASM
**END OF PASS 1

**END OF PASS 2

0801                1           ORG $801
0801                2           OBJ $801
1100                3  FNHI     EQU $1100
1200                4  FNLO     EQU $1200
1300                5  TRACK    EQU $1300
1400                6  SECTOR   EQU $1400
00FA                7  PBHI     EPZ $FA
00F9                8  PBLO     EPZ $F9
00FC                9  SBHI     EPZ $FC
00FB               10  SBLO     EPZ $FB
00FE               11  CFILE    EPZ $FE
00FF               12  TFILE    EPZ $FF
DB3A               13  STROUT   EQU $DB3A
0025               14  CV       EPZ $25
0024               15  CH       EPZ $24
0022               16  WNDTOP   EPZ $22
0023               17  WNDBTM   EPZ $23
FD0C               18  RDKEY    EQU $FD0C
03D9               19  RWTS     EQU $3D9
0200               20  IN       EQU $200
FDED               21  COUT     EQU $FDED
0897               22  LBYTE    EQU $897
0898               23  HBYTE    EQU $898
FB2F               24  INIT     EQU $FB2F
FC58               25  HOME     EQU $FC58
FCA8               26  WAIT     EQU $FCA8
C030               27  SPEAKER  EQU $C030
FE80               28  SETINV   EQU $FE80
FE84               29  SETNORM  EQU $FE84
C000               30  KEYBD    EQU $C000
FD28               31  CLKEY    EQU $FD28
FDDA               32  PRBYTE   EQU $FDDA
FD6F               33  GETLN1   EQU $FD6F
FF3A               34  BELL     EQU $FF3A
FC24               35  VTABZ    EQU $FC24
0028               36  BASL     EPZ $28
0029               37  BASH     EPZ $29
002A               38  BAS2L    EPZ $2A
002B               39  BAS2H    EPZ $2B
FC70               40  SCROLL   EQU $FC70
0801 4C B7 08      41           JMP SETUP
0804 00 00 00      42           HEX 00000000000000000000
0807 00 00 00
080A 00 00 00
080D 00
080E 00 00         43           HEX 0000
0810               44  ;
0810               45  ;IOB
0810               46  ;
0810 01 60 01      47           HEX 01600100
0813 00
0814 11 0F 21      48           HEX 110F2108
0817 08
0818 00 15 00      49           HEX 00150000
081B 00
081C 01 00 FE      50           HEX 0100FE60
081F 60
0820 01            51           HEX 01
0821               52  ;
0821               53  ;DEVICE CHARACTERISTIC TABLE
0821               54  ;
0821 00 01 EF      55           HEX 0001EFD8
0824 D8
0825               56  ;
0825               57  ;STRING
0825               58  ;
0825 CD C1 C3      59           ASC "MACHINE CODES
                                ASC "OFFSETTER   (C) 11 SUI KAY"
0828 C8 C9 CE
082B C5 A0 C3
082E CF C4 C5
0831 D3 A0 CF
0834 C6 C6 D3
0837 C5 D4 D4
083A C5 D2 A0
083D A0 A0 A8
0840 C3 A9 A0
0843 CC C9 A0
0846 D3 C9 D5
0849 A0 CB C1
084C D9
084D 00            60           HEX 00
084E C6 C9 CC      61           ASC "FILENAME: "
0851 C5 CE C1
0854 CD C5 BA
0857 A0
0858 00            62           HEX 00
0859 C1 C4 C4      63           ASC "ADDRESS ORGINAI $"
085C D2 C5 D3
085F D3 A0 CF
0862 D2 C7 C9
0865 CE C1 CC
0868 A0 A4
086A 00            64           HEX 00
086B A0 A0 A0      65           ASC "        TARGET  $"
086E A0 A0 A0
0871 A0 A0 D4
0874 C1 D2 C7
0877 C5 D4 A0
087A A0 A4
087C 00            66           HEX 00
087D 87            67           HEX 87
087E CE CF A0      68           ASC "NO BINARY FILE AVAILABLE"
0881 C2 C9 CE
0884 C1 D2 D9
0887 A0 C6 C9
088A CC C5 A0
088D C1 D6 C1
0890 C9 CC C1
0893 C2 CC C5
0896 00 00 00      69           HEX 000000
0899 87            70           HEX 87
089A C9 CE D3      71           ASC "INSERT DISK AND PRESS A KEY "
089D C5 D2 D4
08A0 A0 C4 C9
08A3 D3 CB A0
08A6 C1 CE C4
08A9 A0 D0 D2
08AC C5 D3 D3
08AF A0 C1 A0
08B2 CB C5 D9
08B5 A0
08B6 00            72           HEX 00
08B7               73  ;
08B7               74  ;SET UP
08B7               75  ;
08B7 20 84 FE      76  SETUP    JSR SETNORM
08BA 20 2F FB      77           JSR INIT
08BD 20 58 FC      78           JSR HOME
08C0 20 30 0B      79           JSR PRLINE
08C3 A0 08         80           LDY #$08
08C5 A9 25         81           LDA #$25
08C7 20 3A DB      82           JSR STROUT
08CA 20 30 0B      83           JSR PRLINE
08CD A9 03         84           LDA #$03
08CF 85 22         85           STA WNDTOP
08D1 A9 0C         86           LDA #$0C
08D3 85 25         87           STA CV
08D5 20 57 0B      88           JSR VTAB
08D8 A9 99         89           LDA #$99
08DA 20 3A DB      90           JSR STROUT
08DD 20 0C FD      91           JSR RDKEY
08E0 20 58 FC      92           JSR HOME
08E3 A9 00         93           LDA #$00
08E5 85 FB         94           STA SBLO
08E7 85 F9         95           STA PBLO
08E9 A9 01         96           LDA #$01
08EB 85 FF         97           STA TFILE
08ED A9 20         98           LDA #$20
08EF 85 FC         99           STA SBHI
08F1 A9 15        100           LDA #$15
08F3 85 FA        101           STA PBHI
08F5 A9 11        102           LDA #$11
08F7 8D 14 08     103           STA $814
08FA A9 0F        104           LDA #$0F
08FC 8D 15 08     105           STA $815
08FF AD F7 B7     106           LDA $B7F7
0902 8D 11 08     107           STA $811          ;SLOT #
0905 AD F8 B7     108           LDA $B7F8
0908 8D 12 08     109           STA $812          ;DRIVE #
090B A9 08        110  LB5      LDA #$08
090D A0 10        111           LDY #$10
090F 20 D9 03     112           JSR RWTS
0912 A9 0B        113           LDA #$0B
0914 85 F9        114           STA PBLO
0916 A0 00        115  LB4      LDY #$00
0918 B1 F9        116           LDA (PBLO),Y
091A C9 FF        117           CMP #$FF
091C D0 07        118           BNE LB151
091E E6 F9        119           INC PBLO
0920 E6 F9        120           INC PBLO
0922 4C 65 09     121           JMP LB1
0925 C9 00        122  LB151    CMP #$00
0927 F0 4A        123           BEQ START
0929 A6 FF        124           LDX TFILE
092B 9D 00 13     125           STA TRACK,X
092E E6 F9        126           INC PBLO
0930 B1 F9        127           LDA (PBLO),Y
0932 9D 00 14     128           STA SECTOR,X
0935 E6 F9        129           INC PBLO
0937 B1 F9        130           LDA (PBLO),Y
0939 29 07        131           AND #$07
093B C9 04        132           CMP #$04
093D D0 26        133           BNE LB1
093F E6 F9        134           INC PBLO
0941 B1 F9        135  LB33     LDA (PBLO),Y
0943 91 FB        136           STA (SBLO),Y
0945 C8           137           INY
0946 C0 1E        138           CPY #$1E
0948 D0 F7        139           BNE LB33
094A A9 00        140           LDA #$00
094C 91 FB        141           STA (SBLO),Y
094E C6 F9        142           DEC PBLO
0950 A5 FC        143           LDA SBHI
0952 9D 00 11     144           STA FNHI,X
0955 A5 FB        145           LDA SBLO
0957 9D 00 12     146           STA FNLO,X
095A 18           147           CLC
095B 69 20        148           ADC #$20
095D 85 FB        149           STA SBLO
095F D0 02        150           BNE LB31
0961 E6 FC        151           INC SBHI
0963 E6 FF        152  LB31     INC TFILE
0965 A5 F9        153  LB1      LDA PBLO
0967 18           154           CLC
0968 69 21        155           ADC #$21
096A 85 F9        156           STA PBLO
096C D0 A8        157           BNE LB4
096E CE 15 08     158           DEC $815
0971 D0 98        159           BNE LB5
0973              160  ;
0973              161  ;PRINT HEADING
0973              162  ;
0973 A9 13        163  START    LDA #$13
0975 85 25        164           STA CV
0977 20 57 0B     165           JSR VTAB
097A 20 30 0B     166           JSR PRLINE
097D C6 FF        167           DEC TFILE
097F D0 03        168           BNE LB8
0981 4C 1B 0B     169           JMP EMPTY
0984 C6 FF        170  LB8      DEC TFILE
0986 A2 00        171           LDX #$00
0988 86 FE        172           STX CFILE
098A A9 03        173           LDA #$03
```

起始地址搬移器

```
098C 85 25      174          STA CV
098E A9 13      175          LDA #$13
0990 85 23      176          STA WNDBTM
0992 20 3B 0B   177 LB10     JSR PRFILE
0995 E6 25      178          INC CV
0997 A5 FE      179          LDA CFILE
0999 C5 FF      180          CMP TFILE
099B F0 08      181          BEQ LB9
099D E6 FE      182          INC CFILE
099F A5 25      183          LDA CV
09A1 C9 13      184          CMP #$13
09A3 D0 ED      185          BNE LB10
09A5 A9 03      186 LB9      LDA #$03
09A7 85 25      187          STA CV
09A9 A9 00      188          LDA #$00
09AB 85 FE      189          STA CFILE
09AD A0 20      190 LB13     LDY #$20
09AF A9 08      191 LB17     LDA #$08
09B1 20 A8 FC   192          JSR WAIT
09B4 AD 30 C0   193          LDA SPEAKER
09B7 88         194          DEY
09B8 D0 F5      195          BNE LB17
09BA 20 80 FE   196 LB100    JSR SETINV
09BD 20 3B 0B   197          JSR PRFILE
09C0 20 84 FE   198          JSR SETNORM
09C3 2C 00 C0   199 LB11     BIT KEYBD
09C6 10 FB      200          BPL LB11
09C8 20 3B 0B   201          JSR PRFILE
09CB 20 28 FD   202          JSR CLKEY
09CE C9 8D      203          CMP #$8D
09D0 D0 03      204          BNE LB81
09D2 4C 33 0A   205          JMP SELECT
09D5 C9 91      206 LB81     CMP #$91          ↑^Q
09D7 D0 03      207          BNE CHOICE
09D9 4C 12 0B   208          JMP END
09DC C9 88      209 CHOICE   CMP #$88
09DE F0 1D      210          BEQ LB20
09E0 C9 95      211          CMP #$95          ↓DOWN
09E2 D0 D6      212          BNE LB100
09E4            213 ;
09E4            214 ;DOWN
09E4            215 ;
09E4 A5 FE      216          LDA CFILE
09E6 C5 FF      217          CMP TFILE
09E8 F0 C3      218 LB21     BEQ LB13
09EA E6 FE      219          INC CFILE
09EC A5 25      220          LDA CV
09EE C9 12      221          CMP #$12
09F0 D0 06      222          BNE LB30
09F2 20 70 FC   223          JSR SCROLL
09F5 4C BA 09   224          JMP LB100
09F8 E6 25      225 LB30     INC CV
09FA 4C BA 09   226          JMP LB100
09FD            227 ;
09FD            228 ;UP
09FD            229 ;
09FD A5 FE      230 LB20     LDA CFILE
09FF F0 E7      231          BEQ LB21
0A01 C6 FE      232          DEC CFILE
0A03 A5 25      233          LDA CV
0A05 C9 03      234          CMP #$03
0A07 D0 25      235          BNE LB40
0A09 A9 12      236          LDA #$12
0A0B 48         237          PHA
0A0C 20 24 FC   238          JSR VTABZ
0A0F A5 28      239 LB70     LDA BASH
0A11 85 2A      240          STA BAS2H
0A13 A5 29      241          LDA BASH
0A15 85 2B      242          STA BAS2H
0A17 A0 27      243          LDY #$27
0A19 68         244          PLA
0A1A E9 00      245          SBC #$00
0A1C C9 02      246          CMP #$02
0A1E F0 10      247          BEQ LB50
0A20 18         248          CLC
0A21 48         249          PHA
0A22 20 24 FC   250          JSR VTABZ
0A25 B1 28      251 LB60     LDA (BASL),Y
0A27 91 2A      252          STA (BAS2L),Y
0A29 88         253          DEY
0A2A 10 F9      254          BPL LB60
0A2C 30 E1      255          BMI LB70
0A2E C6 25      256 LB40     DEC CV
0A30 4C BA 09   257 LB50     JMP LB100
0A33            258 ;
0A33            259 ;SELECT
0A33            260 ;
0A33 A5 25      261 SELECT   LDA CV
0A35 85 FD      262          STA $FD
0A37 A9 18      263          LDA #$18
0A39 85 23      264          STA WNDBTM
0A3B A9 14      265          LDA #$14
0A3D 85 25      266          STA CV
0A3F 20 57 0B   267          JSR VTAB
0A42 A9 4E      268          LDA #$4E
0A44 20 3A DB   269          JSR STROUT
0A47 A6 FE      270          LDX CFILE
0A49 BD 01 12   271          LDA FNLO+1,X
0A4C BC 01 11   272          LDY FNHI+1,X
0A4F 20 3A DB   273          JSR STROUT
0A52 A6 FE      274          LDX CFILE
0A54 BD 01 13   275          LDA TRACK+1,X
0A57 8D 14 08   276          STA $814
0A5A BD 01 14   277          LDA SECTOR+1,X
0A5D 8D 15 08   278          STA $815
0A60 A9 08      279          LDA #$08
0A62 A0 10      280          LDY #$10
0A64 20 D9 03   281          JSR RWTS
0A67 AD 0C 15   282          LDA $150C
0A6A 8D 14 08   283          STA $814
0A6D AD 0D 15   284          LDA $150D
0A70 8D 15 08   285          STA $815
0A73 A9 08      286          LDA #$08
0A75 A0 10      287          LDY #$10
0A77 20 D9 03   288          JSR RWTS

0A7A A0 08      289          LDY #$08
0A7C A9 59      290          LDA #$59
0A7E 20 3A DB   291          JSR STROUT
0A81 AD 01 15   292          LDA $1501
0A84 20 DA FD   293          JSR PRBYTE
0A87 AD 00 15   294          LDA $1500
0A8A 20 DA FD   295          JSR PRBYTE
0A8D A2 05      296          LDX #$05
0A8F 20 4A F9   297          JSR $F94A         ;BLANK
0A92 A0 08      298          LDY #$08
0A94 A9 67      299          LDA #$67
0A96 20 3A DB   300          JSR STROUT        ;"I S"
0A99 AD 03 15   301          LDA $1503
0A9C 20 DA FD   302          JSR PRBYTE
0A9F AD 02 15   303          LDA $1502
0AA2 20 DA FD   304          JSR PRBYTE
0AA5 E6 25      305          INC CV
0AA7 20 57 0B   306 ERROR    JSR VTAB
0AAA A9 6B      307          LDA #$6B
0AAC 20 3A DB   308          JSR STROUT
0AAF 20 6F FD   309          JSR GETLN1
0AB2 A9 00      310          LDA #$00
0AB4 8D 97 08   311          STA LBYTE
0AB7 8D 98 08   312          STA HBYTE
0ABA A0 00      313          LDY #$00
0ABC B9 00 02   314 NXTNIB   LDA IN,Y
0ABF C9 8D      315          CMP #$8D
0AC1 F0 27      316          BEQ RTS1
0AC3 49 B0      317          EOR #$B0
0AC5 C9 0A      318          CMP #$0A
0AC7 90 0E      319          BCC DIG
0AC9 69 88      320          ADC #$88
0ACB C9 FA      321          CMP #$FA
0ACD B0 08      322          BCS DIG
0ACF 20 3A FF   323          JSR BELL
0AD2 C6 25      324          DEC CV
0AD4 4C A7 0A   325          JMP ERROR
0AD7 A2 03      326 DIG      LDX #$03
0AD9 0A         327          ASL
0ADA 0A         328          ASL
0ADB 0A         329          ASL
0ADC 0A         330          ASL
0ADD 0A         331 NXTBIT   ASL
0ADE 2E 97 08   332          ROL LBYTE
0AE1 2E 98 08   333          ROL HBYTE
0AE4 CA         334          DEX
0AE5 10 F6      335          BPL NXTBIT
0AE7 C8         336          INY
0AE8 D0 D2      337          BNE NXTNIB
0AEA AD 97 08   338 RTS1     LDA LBYTE
0AED 8D 00 15   339          STA $1500
0AF0 AD 98 08   340          LDA HBYTE
0AF3 8D 01 15   341          STA $1501
0AF6 A9 02      342          LDA #$02
0AF8 8D 1C 08   343          STA $81C
0AFB A9 08      344          LDA #$08
0AFD A0 10      345          LDY #$10
0AFF 20 D9 03   346          JSR RWTS
0B02 A9 01      347          LDA #$01
0B04 8D 1C 08   348          STA $81C
0B07 A5 FD      349          LDA $FD
0B09 85 25      350          STA CV
0B0B A9 13      351          LDA #$13
0B0D 85 23      352          STA WNDBTM
0B0F 4C AD 09   353          JMP LB13
0B12 20 2F FB   354 END      JSR INIT
0B15 20 58 FC   355          JSR HOME
0B18 4C 3C D4   356          JMP $D43C         ;BASIC
0B1B            357 ;
0B1B            358 ;SUBROUTINE EMPTY
0B1B            359 ;
0B1B 20 58 FC   360 EMPTY    JSR HOME
0B1E A9 0B      361          LDA #$0B
0B20 85 25      362          STA CV
0B22 20 57 0B   363          JSR VTAB
0B25 A9 7D      364          LDA #$7D
0B27 20 3A DB   365          JSR STROUT
0B2A 20 0C FD   366          JSR RDKEY
0B2D 4C B7 08   367          JMP SETUP
0B30            368 ;
0B30            369 ;SUBROUTINE PRLINE
0B30            370 ;
0B30 A2 28      371 PRLINE   LDX #$28
0B32 A9 AD      372 LPR      LDA #$AD
0B34 20 ED FD   373          JSR COUT
0B37 CA         374          DEX
0B38 D0 F8      375          BNE LPR
0B3A 60         376          RTS
0B3B            377 ;
0B3B            378 ;SUBROUTINE PRFILE
0B3B            379 ;
0B3B A9 00      380 PRFILE   LDA #$00
0B3D 85 24      381          STA CH
0B3F 20 22 FC   382          JSR $FC22         ;VTAB
0B42 A6 FE      383          LDX CFILE
0B44 BD 01 12   384          LDA FNLO+1,X
0B47 BC 01 11   385          LDY FNHI+1,X
0B4A 20 3A DB   386          JSR STROUT
0B4D A2 09      387          LDX #$09
0B4F 20 4A F9   388          JSR $F94A         ;BLANK
0B52 A0 27      389          LDY #$27
0B54 91 28      390          STA (BASL),Y
0B56 60         391          RTS
0B57            392 ;
0B57            393 ;VTAB
0B57            394 ;
0B57 A9 00      395 VTAB     LDA #$00
0B59 85 24      396          STA CH
0B5B 20 22 FC   397          JSR $FC22         ;VTAB
0B5E A0 08      398          LDY #$08
0B60 60         399          RTS
0B61            400          END

***** END OF ASSEMBLY
```

配有特殊音响效果的TRANSYLVINA

怎樣把MKB的說話部份加進自己的程式內

筆者上一期編寫的WILL 'O THE WISP 附有說話版本，不過必須利用有SPEECH CHIP 的MKB 而且要兩個磁碟機才會說話。主要的原因是擁有SPEECH CHIP 的讀者不會太多，相信和價錢昂貴有關，一枚SPEE CH CHIP 大概售二百多元；而說話部份資料必須從另外一隻磁碟輸入。原本遊戲的磁碟載有超過100 幅圖畫，再加上聲音效果資料等已經沒有多餘地方可利用。原則上可以編排兩隻資料磁碟交替運用，這樣便不用兩部磁碟機了，但筆者相信擁有SPEECH 的朋友大多已有兩部，所以編寫時利用直接兩部磁碟機可以無須把磁碟取出又放回以節省時間。

一般動作遊戲利用說話不會要求太多，所以通常都不會加入說話部份。就算間中加入，都只是很短的幾句話而已。但一些ADVENTURE 或WAR GAME ，對說話要求必會增加，這些遊戲通常要求玩者直接與電腦交談，電腦能夠說話自然大大增強效果了。試想假如一個探險遊戲是完全沉默的話，這是多麼乏味呢？

事實上，以MKB產生說話只倚靠一個COMPOSITE DRIVER。該程式只共佔用154個BYTES，可以說對記憶體要求甚少，我們只須預留大約一個PAGE 左右空間存放說話資料，所以一般BASIC 都能夠發揮說話功能。可是假如我們需要產生大量的談話資料又如何呢？只要從磁碟輸入每一句談話便可。COMPOSITE DRIVE 可以同一時間處理255個字母以內的句語，一般的句子已經可以應付了。（我們不會不斷地說話而不停止吧）以255個字母來說，一句說話只佔用 2 至 3 個SECTOR ，一隻磁碟可以把一篇很長的文章讀出來了。

程式的LISTING 是COMPOSITE DRIVER的原始碼。原來MKB 手冊上列出的COMPOSITE DRIVER 錯誤甚多，令筆者花了一段時間去整理，現在列出的是一個完整無誤的程式，讀者可以隨時運用。

未討論如何把說話產生之前，首先看看如何編制說話的資料。

編制一句說話的資料要利用RULE EDITOR 處理，筆者建議採用MOCKINGBOARD TOOL KIT（MTK） 內的RULE EDITOR ，因爲這是一個發音比較準確的EDIT－OR，無須花費時間去EDIT 發音部份。BOOT 起該磁碟然後待CATALOG 出現後鍵入RUN DEMO， 螢幕會顯示MKB 的目前狀況，再按CTRL－RESET 然後鍵入BRUN RULE EDITOR，該程式便會輸入。這裏有幾個COMMA－ND 要留意如下：

T　輸入句語文字讓MKB 讀出
N　消去句語文字讓另外一句輸入
R　重新再讀句語

> 編者註：電腦時代長期訂戶（連同磁碟訂閱者）將可免費獲得麥法基今期改編的加插有聲响效果的TRANSYLVI－NA ADVENTURE 遊戲。請携同磁碟到讀者服務部CO－PY ，或利用郵寄方式索取，郵寄時請附同一個貼上一元郵票的回郵信封。至於MBC、AFC、電腦時代／微電腦技術／COMPUSOFT 磁碟月刊長期訂戶則可用15元購買此遊戲。除上述以外的讀者，對不起，我們不能售與你。

E　EDIT 發音
U　UPDATE 發音
∧S 把說話資料存放到磁碟上　∧I 處理說話高低音
∧Z 返回RULE EDITOR　∧R 處理說話快慢
∧A 處理聲量控制　∧F 處理說話聲調

（∧表示要先按CTRL 鍵）

上面所列是比較常用的指令，還有其他暫時不去理會。當RULE EDITOR 輸入後首先按A鍵然後按T鍵，螢幕會顯示文字部份讓我們把要讀的句字打入。假設我們打入THE RAIN IN SPAIN DROPS MAINLY ON THE PLAIN（RETURN），電腦會立即把整句讀出來，然後螢幕下面會把該句說話資料以BINARY CODE 列出，這便是COM－POSITE DRIVE 所需的資料了。

事實上，現在讀出的句話毫無感情可言，因爲每個字聲調都一樣。倘若我們按N鍵把這句消去，然後重新打入THE ／RAIN／IN S／PAIN DROPS MAINLY／ON THE ／PLAIN／ 則讀出的句語比較完善，所有兩條斜綫之內的字都以較高聲調讀出，而變得有生氣了。

很多時我們需要把句子作出少許整理讓句子讀出時有準確的發音，鍵N消去句子然後鍵入YOU ARE IN A DARK AREA OF THE FOREST。倘若沒有文字顯示，我們根本不知道它說些甚麼，除了沒加入聲調指示外，這個FOR E－ST 發音不準確。重新打入如下

YOU ARE IN A ／DARK AR／EA OF THE ／FOR／ EST 這時整句句子被準確地說出來了，因爲FOREST 這個字被分列爲兩個發音。

再看看鍵入PIFFERENCE 這個字，這是一個完全發音不準確的字，因爲原來這字中間的DIFF E RENCE 的FER 部份不應該被讀出來，但RULE EDITOR 却把它讀如DIFF FER RENCE，所以鍵入時把它打成 ／DIFF ／RENCE 把中開的E 音刪去。

又例如SOUTHERN 這個字亦不準確，必須被分列鍵入如SOUTH ERN。

怎樣把 MKB 的說話部份加進自己的程式內

假如一句句子被準確讀出後，我們便須按 ^S 把它存放到磁碟上，當然要加上一個FILE NAME。

要把句子讓自己的程式讀出來很簡單，首先把COMPOSITE DRIVE 輸入電腦，要留意輸入的起始位置，我們可以把該程式組合在任何位址上，假設是$6D00（27904）而說話資料可以放在其他空間，例如$8900（35072），但要和其他程式所用到。下面是一個例子：

```
10  D$=CHR$(4)
20  PRINT D$"BLOAD PHRASE,A35072"
30  POKE 249,0:POKE 250,137
40  CALL 27904
50  IF PEEK(255)>0 THEN 50
60  END
```

其中第30行的249和250是說話資料位址一對指位器，以低高位元存入（$8900是$00和$89，即00和137）。第40行呼叫COMPOSITE DRIVER把句子讀出。而第50行的255是判斷句子是否還在進行說話，大於零表示說話仍在進行，所以不斷於50行循環直至255等於零時才繼續程式其他部份，這樣我們不會干擾說話。

其他部份下次再談，這一期筆者把PENGIUM 出品的一個名爲TRANSYLVINA 的恐怖 ADVENTURE 配上 MOCKINGBOARD 各種效果，有興趣的讀者可以向電腦時代詢問。■

```
                        1               ORG   $6D00
                        2   *
                        3   TABPTR      EQU   $F9
                        4   OUTPTR      EQU   $FB
                        5   ENDPTR      EQU   $FD
                        6   BUSY        EQU   $FF
                        7   IRQL        EQU   $3FE
                        8   IRQH        EQU   $3FF
                        9   BASE        EQU   $C440
                       10   DURPHON     EQU   BASE
                       11   INFLECT     EQU   BASE+$01
                       12   RATEINF     EQU   BASE+$02
                       13   CTTRAMP     EQU   BASE+$03
                       14   FILFREQ     EQU   BASE+$04
                       15   PCR         EQU   $C48C
                       16   IFR         EQU   $C48D
                       17   IER         EQU   $C48E
                       18   *
                       19   *
                       20   *
                       21   *
                       22   * SETUP ROUTINE
                       23   *
6D00: 78               24               SEI
6D01: A9 55            25               LDA   #<INTERR
6D03: 8D FE 03         26               STA   IRQL
6D06: A9 6D            27               LDA   #>INTERR
6D08: 8D FF 03         28               STA   IRQH
6D0B: A9 00            29               LDA   #$00
6D0D: 8D 03 C4         30               STA   $C403
6D10: 8D 02 C4         31               STA   $C402
6D13: A5 FA            32               LDA   TABPTR+1
6D15: 85 FC            33               STA   OUTPTR+1
6D17: A6 F9            34               LDX   TABPTR
6D19: E8               35               INX
6D1A: E8               36               INX
6D1B: D0 02            37               BNE   CONT
6D1D: E6 FC            38               INC   OUTPTR+1
6D1F: 86 FB            39   CONT        STX   OUTPTR
6D21: A0 01            40               LDY   #$01
6D23: B1 F9            41               LDA   (TABPTR),Y
6D25: 18               42               CLC
6D26: 65 FA            43               ADC   TABPTR+1
6D28: 85 FE            44               STA   ENDPTR+1
6D2A: 88               45               DEY
6D2B: B1 F9            46               LDA   (TABPTR),Y
6D2D: 18               47               CLC
6D2E: 65 F9            48               ADC   TABPTR
6D30: 90 02            49               BCC   CONTI
6D32: E6 FE            50               INC   ENDPTR+1
6D34: 85 FD            51   CONTI       STA   ENDPTR
                       52   *
                       53   *
6D36: A9 FF            54   CONT5       LDA   #$FF
6D38: 85 FF            55               STA   BUSY
6D3A: A9 0C            56               LDA   #$0C
6D3C: 8D 8C C4         57               STA   PCR
6D3F: A9 80            58               LDA   #$80
6D41: 8D 43 C4         59               STA   CTTRAMP
6D44: A9 C0            60               LDA   #$C0
6D46: 8D 40 C4         61               STA   DURPHON
6D49: A9 70            62               LDA   #$70
6D4B: 8D 43 C4         63               STA   CTTRAMP
6D4E: A9 82            64               LDA   #$82
6D50: 8D 8E C4         65               STA   IER
6D53: 58               66               CLI
6D54: 60               67               RTS
                       68   *
                       69   * INTERRUPT ROUTINE
6D55: 8A               70   INTERR      TXA
6D56: 48               71               PHA
6D57: 98               72               TYA
6D58: 48               73               PHA
6D59: A9 02            74               LDA   #$02
6D5B: 8D 8D C4         75               STA   IFR
6D5E: A0 00            76               LDY   #$00
6D60: A2 04            77               LDX   #$04
6D62: A5 FB            78               LDA   OUTPTR
6D64: C5 FD            79               CMP   ENDPTR
6D66: 90 20            80               BCC   CONT6
6D68: A5 FC            81               LDA   OUTPTR+1
6D6A: C5 FE            82               CMP   ENDPTR+1
6D6C: 90 1A            83               BCC   CONT6
6D6E: A9 00            84               LDA   #$00
6D70: 8D 40 C4         85               STA   DURPHON
6D73: A9 70            86               LDA   #$70
6D75: 8D 43 C4         87               STA   CTTRAMP
6D78: A9 00            88               LDA   #$00
6D7A: 85 FF            89               STA   BUSY
6D7C: A9 02            90               LDA   #$02
6D7E: 8D 8E C4         91               STA   IER
6D81: 68               92   RET         PLA
6D82: A8               93               TAY
6D83: 68               94               PLA
6D84: AA               95               TAX
6D85: A5 45            96               LDA   $45
6D87: 40               97               RTI
                       98   *
6D88: B1 FB            99   CONT6       LDA   (OUTPTR),Y
6D8A: 9D 40 C4        100               STA   BASE,X
6D8D: E6 FB           101               INC   OUTPTR
6D8F: D0 02           102               BNE   CONT7
6D91: E6 FC           103               INC   OUTPTR+1
6D93: CA              104   CONT7       DEX
6D94: E0 FF           105               CPX   #$FF
6D96: D0 F0           106               BNE   CONT6
6D98: F0 E7           107               BEQ   RET
                      108   *

--End assembly--

154 bytes

Errors: 0
```

程式編寫入門之四 EXEC指令的用途

▌黃兆才▐

DOS指令EXEC，它與RUN或BRUN相似，但它所執行的並不是A，I或B型的檔案程式，而是文字檔案（TEXT FILE）。文字檔案內所儲存的亦不是一般的資料，而是一些指令或程式行（PROGRAM LINES）。

至於EXEC的用途可分三類：

第一類：將APPLESOFT BASIC程式轉爲整數BAS－IC，或將整數BASIC轉爲APPLESOFT BASIC。方法是先將欲轉變之程式LOAD入電腦，再鍵入LISTING 1（加至程式之內），然後用RUN來執行程式，程式是會將一個名爲TRANSFER的文字檔案寫到磁碟內，跟着按FP或INT進入欲轉爲的語言系統，再鍵入EXEC TRANSF－ER，等候游標的再出現，此時程式已被轉放到另一個語言系統內，只要將程式SAVE回磁碟內便大功告成，此時TRANSFER這個文字檔案亦可功成身退，將其DELETE可也。不過有幾點是要注意的：

①欲轉變之程式的第一行號不能是零，因爲LISTING 1便是以零行號加入至程式內的。

表一

```
0 D$="":PRINT D$;"OPEN TRANSFER":
    PRINT D$;"WRITE TRANSFER":POK
    E 33,33:LIST A,B:PRINT D$;"CL
    OSE":TEXT:END
```

②磁碟必須要有足夠的空間來儲存文字檔案TRANSF－ER。

③APPLESOFT BASIC與整數BASIC兩者並非完全互通的，所以程式內的語句要作適當的修改才能於轉變後被執行。因爲整數BASIC是會先檢查每句的語法結構是否正確才接納，而APPLESOFT BASIC則在執行程式時才作語法結構的檢查，所以欲將APPLESOFT BASIC轉爲整數BASIC前，要先作修改，然後轉變；而欲將整數BASIC轉爲APPLESOFT BASIC，則需先轉變，然後再作修改。至於修改的方法可參考電腦時代第8期之"將INTEGER轉爲APPLESOFT"一文。

④在LISTING 1內D$＝""，在兩個"符號中是鍵入CTRL－D，而LIST A，B內的A是該程式的第一行號，B是該程式的最後一行號。

```
10 D$=CHR$(4):Q$=CHR$(34):PR$=
    "?D$"+Q$:F$=Q$+";FI$"
20 IN$="INPUT"+Q$+"FILENAME?"
    +F$
30 UN$=PR$+"UNLOCK"+F$+":":DE$
    =PR$+"DELETE"+F$+":"
40 SA$=PR$+"SAVE"+F$+":":LO $=P
    R$+"LOCK"+F$
50 PRINT D$"OPEN RENEW":PRINT
    D$"WRITE RENEW"
60 PRINT "D$=CHR$(4):POKE 118,
    0:POKE 43699,0:";IN$;
70 PRINT":POKE 43699,1:";UN$;D
    E$;SA$;LO$
80 PRINT D$"CLOSE"
```

表二

第二類：將兩個或以上的BASIC程式（APPLESOFT或INTEGER）合併爲一。電腦時代經常刊登很多有用的BASIC程式，其中亦不少是非常冗長的，對於不擅於打字的朋友來說便辛苦了。如果你有另一個（或數位）擁有電腦及磁碟機的朋友，便可將你的辛苦分輕。方法便是將程式分爲兩部份（或數部份），分別由各人鍵入及儲於磁碟上；但是只有一位朋友可以將他那部份的程式SAVE到磁碟上，其餘的朋友就必須將他那部份的程式再加入LISTING 1，然後RUN程式以便將該部份的程式轉爲文字檔案存於磁碟上。當全部準備妥當便可進行合併的工作了，首先將SAVE在磁碟上的程式LOAD入電腦，然後將其餘的部份（文字檔案）用EXEC指令便可加到存於電腦內的程式而合爲一個程式，最後SAVE回磁碟上便可。要特別一提的是進行合併工作時是不需順着檔案次序來EXEC的。

```
10 D$=CHR$(4):PRINT D$"OPEN
    STOPCAT":PRINT D$"WRITE STO
    PCAT"
20 PRINT "CALL-151"
30 PRINT "AE39:20 80 B6"
40 PRINT "B680:20 0C FD C9 9B
    D0 03 4C D0 03 60"
50 PRINT "3D0G"
60 PRINT D$"CLOSE"
```

表三

第三類：這是一個較爲少人用的用途，便是將我們時常需要重覆鍵入電腦的一些指令寫成文字檔案，然後於需要時EXEC該檔案，由它來替我們工作，以減少我們指頭的痛苦。例如我經常需要將已寫好的程式（存於磁碟上並已LOCK上）更改，再SAVE回磁碟上，便要做以下四項工作：UN－LOCK，DELETE，SAVE及LOCK；如果利用LISTI－NG 2產生一個名爲RENEW的文字檔案，則只要在更改好程式後EXEC RENEW，跟着鍵入欲取代之程式名稱便可；因爲RENEW這個文字檔案會替我將所有的事情做好，不要我再操心。又例如想利用第21期七項DOS小玩意中的暫停CATALOG功能，而又不想更改磁片上的標準DOS，是不是要每一次BOOT機後都要跟文中所示的方法鍵入呢？那又不必，只要利用LISTING 3便會產生一個名爲STOP－CAT的文字檔案，以後只需EXEC STOPCAT便可。

以上只舉了兩個例子，但是EXEC在這方面的用途是相當大的，希望各位朋友不要將它"冷落"。 ■

新讀者需知

怎樣打入電腦時代程式列表

電腦時代每期刊出的程式列表全部適用於APPLE 系列電腦或兼容電腦。程式列表大致分爲兩類：APPLESOFT BASIC 及機械語言（MACHINE CODE）。另外一類是組編語言（ASSEMBLER），對於初學者或初接觸電腦的人士可以暫時不用理會組編語言是什麼東西。現在讓我們解釋詳述怎樣打入APPLESOFT BASIC 程式列表及機械語言列表。

```
10  REM  RING THE BELL
20  FOR J = 1 TO 5: PRINT  CHR$ (
    7): NEXT J
30  END
```

列表㈠是一個簡單的BASIC 程式

什麼是 BASIC 列表？

列表㈠在執行時（RUN）可令電腦的喇叭發出聲响。在未鍵入列表之前，閣下必須要有下列東西：

A. 一張已FORMAT 的磁碟。

B. 電腦主記憶體內有DOS 3·3 系統。

要打入列表㈠，可依下列步驟做：

1. 鍵入NEW，然後按RETURN 鍵。此舉目的是清除電腦主記憶體的舊資料。
2. 按照列表㈠行句10一模一樣全部鍵入電腦，直至打入完最後一個英文字BELL，然後按RETURN 鍵。
3. 行句20及30亦照一樣方式鍵入。
4. 將一張已FORMAT 的磁碟放入磁碟機，鍵入SAVE BELL 及按RETURN 鍵。此時磁碟上已產生有一個名爲BELL 的檔案。試打入CATALOG及按RETURN 即可看見螢幕出現如圖A的情況。
5. 由於剛才打入的程式依然留在主記憶體裏，故此，若此時鍵入RUN 〔RETURN〕，電腦的喇叭會即時有反應。
6. 關上電腦的電源掣，再開啓。這時主記憶體上將會清除先前打入的程式。若想再一次RUN 這程式，只要打入RUN BELL 〔RETURN〕電腦就會從磁碟機上取出BELL 這檔案及執行，電腦的喇叭會發出聲响。
7. 在鍵入程式時，假如鍵入錯誤字鍵，可按動左箭咀鍵，讓游標（▩）游返至出錯的字母上，鍵入正確的字鍵後，再按右箭咀鍵返回行尾，再按RETU－RN。

```
] CATALOG

DISK VOLUNN 256

                          圖A
A 002 BELL
```

怎樣鍵入機械語言列表？

```
*300.30B

0300- A2 05 20 DD FB CA F0 03
0308- 4C 02 03 60
```

列表㈡是一個機械語言列表

大家要鍵入列表㈡時，可依照下列步驟：

(1) 鍵入CALL－151及按RETU－RN，螢幕上會出現一個〔*〕訊號，表示已進入MON－ITOR SYSTEM。

(2) 打入0300：A2 05 20 DD FB CA F0 03〔RETURN〕

(3) 再打入下一句
0308：4C 02 03 60 〔RE－TURN〕。

(4) 留意在列表㈡出現的位址03－00之後的〔－〕，在鍵入程式時是用〔：〕（冒號）代替，而冒號〔：〕緊接之後的數值A2 是沒有空位的。

(5) 當鍵入完整個列表後，同時鍵下〔CTRL〕及C 兩個鍵，再按RETURN，電腦會從MON－ITOR 系統返回BASIC 系統，螢幕會出現一個"]"訊號。

(6) 要將先前打入的機械語言列表存入磁碟，必須要用BSAVE 這個指令（與APPLESOFT BASIC 不同，後者是用SAVE），然後加上一個檔案名字（FILE NAME），尾隨一個開始位置值（A$XXXX）及一個檔案長度值（L$XXXX），列表二的例子是：
BSAVE RINGER，A$300，L$C RINGER 即檔案名字，A$後的300 是開始位置值，L$後的C是檔案長度。

(7) 大家可發覺到在電腦時代刊出步驟，將列表㈡左邊的機械語言打入電腦，用同樣指令存入磁碟亦可。有關該列表的名字、開首地址及檔案長度，大家可在文章內找到。

磁碟操作系統

磁碟操作系統是電腦與磁碟機溝通的媒介，電腦時代通常提及的是DOS 3·3 及PRODOS 系統，後者對於初學者比較複雜一些，可以暫時撇開不理。前者DOS 3·3 是最常見的系統，大家只要BOOT 起任何一張沒有加上保護（例如電腦時代每期出版的程式磁碟）即可取得。

上述只是一個很簡單的介紹，假如初學者想知更多更詳細的資料，應參考買機時附隨的三本使用手冊。

```
*300L

0300-  A2 05      LDX   #$05
0302-  20 DD FB   JSR   $FBDD
0305-  CA         DEX
0306-  F0 03      BEQ   $030B
0308-  4C 02 03   JMP   $0302
030B-  60         RTS
```

列表三

電腦時代程式磁碟

由於電腦時代每期都刊登十數個以上的程式，假如一一鍵入，可能會花費許多時間，因此，我們每期出版電腦時代的同時，都會出版一張以該期爲編號的程式磁碟，內裏收錄有所有該期的程式，磁碟大部份是兩面禮錄。

使用程式磁碟方法十分簡單：

(1) 關掉電腦電源

(2) 將程式磁碟放入磁碟機

(3) 開啓電腦電源掣

(4) 磁碟機上的LED 燈會亮着

(5) 螢幕會出現有「電腦時代」的標誌，然後會出現一系列的檔案名稱。

(6) 待一個閃動的方框訊號出現在"]"的右旁時，大家打入RUN XXXX 或BRUN XXXX（如果檔案是APPLESOFT BASIC 即用RUN，如果是機械語言則用BRUN），XXXX 代表著欲執行的檔案名稱。

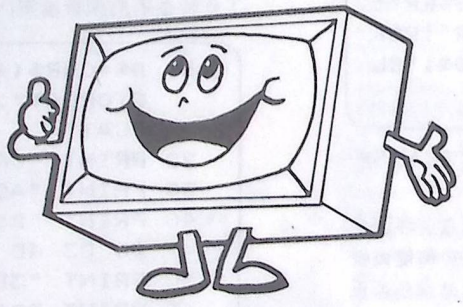
(7) 有時，某些程式會包括有多個檔案，此時，大家必定要跟隨書上所說執行所指定的檔案，假如不照指示，電腦隨時會出現HANG 機。如出現此情況，最佳莫如關閉電腦電源掣，再從頭開始一遍。

的機械語言列表開首有一個用星號"*"圍著的長框，內裏列出有該列表的檔案名字，開首位置及檔案長度，以列表㈡爲例，方框應如出現如下：

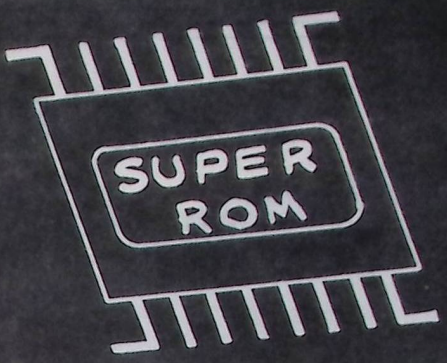
```
10  REM  ++++++++++++++++
15  REM  +   RINGER    +
20  REM  +  A$300,L$C  +
30  REM  ++++++++++++++++
```

(8) 當大家要執行此RINGER程式時，可鍵入BRUN RINGER 即可。大家不妨試做。

(9) 你們會發覺電腦時代上可能出現有如表㈢的列表，大家暫時不用理會它，列表㈢是所謂組編語言列表，是提供給較高程度的電腦用家。對於你們初學者是會感到一頭霧水。不過，如果我們沒有刊出如列表㈡的程式列表而用列表㈢般的形式列出，大家不用担心，只要照前述怎樣打入機械語言列表的



SUPER ROM



超級 ROM 功能比美 WIDE CARD 價值便宜 66%

裝置方法：直接插在 F8 或配上擴展咭，兩者方法皆可。

適合任何蘋果電腦，絕不影響時下軟件的操作功能：

(1)增加三項RESET鍵功能，電腦發燒友必定會喜歡；
包括：按入 RESET 鍵時可以選擇進入監督程式中，或將累加器、X、Y暫存器的內容存起再進入監督程式之內又或進行普通的 RESET。拆碟的朋友必定會喜歡。

(2)改進反組譯機械碼功能，將每次只列出 20 個指令數目改良至可連續不斷地列出，使在印字機輸出時，不會一段段地分開。

(3)尋找機械碼命令，使各位可以像找尋 BASIC字句般把你要找的機械碼一一找出來。

此外尚有其他多項功能，不能盡錄。

附隨一份詳細使用說明書，只售 $35，歡迎到讀者服務部參觀示範。

超級拆碟ROM 接受郵函，請填寫清楚地址及加付郵費／處理費5元，（每多一枚加付一元），掛號加付5元。
配合SUPER ROM 使用的擴展咭經已設計好，並在電腦時代第廿三期小型製作專欄內刊登，歡喜親自動手的讀者可自行製造。或者到讀者服務部購買成品。成品（連JUMPER）訂價45元，底板每塊則售10元。
郵購支票抬頭請寫上「電腦時代出版社」。SUPER ROM連擴展咭一套則售75元。

電腦時代讀者服務部

地址：九龍旺角上海街 438-444 號同珍商業中心 11 字樓 1104 室
電話：3－312007

新增加兩大有獎比賽項目

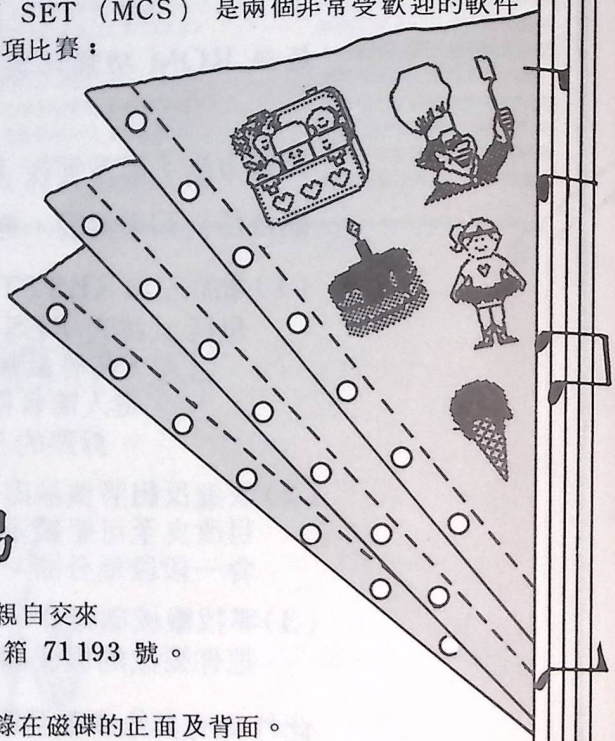
PS圖案設計及MCS作曲比賽

第一項 PRINTSHOP 圖案設計比賽　　第二項 MCS 作曲大賽

PRINTSHOP （PS） 及 MUSIC CONSTRUCTION SET （MCS） 是兩個非常受歡迎的軟件。爲了進一步加強 PS 及 MCS 的用途，我們特別舉辦兩項比賽：

擁有 PS 的讀者可利用 PS 內的 GRAPHIC EDITOR 或我們於電腦時代第廿三期刊出的「專爲 PS 而設的製圖工具」程式來設計一些別出心裁的圖案供 PS 使用。圖案主題不限，最好能有創意。所有入選的作品，每一個可獲高級磁碟一張，若果有十個入選，便可獲得十張磁碟。

至於 MCS 作曲比賽，大家可運用 MCS 來創作歌曲以便供 MOCKING BOARD 播送。大家可自行創作或改編流行歌曲，取材不限。每首入選作品均可獲得任何一期的電腦時代程式磁碟。假如參賽作品能配有動畫顯示，更可獲得額外獎品。



獎品豐富，參加辦法容易

(1) 所有參賽作品，必須收錄在磁碟上寄來下列地址或親自交來電腦時代讀者服務部。郵寄地址：九龍中央郵政信箱 71193 號。信封面必須註明「參加 PS/MCS 比賽」。

(2) PS 圖案及 MCS 的 MUSIC DATA 必須分開收錄在磁碟的正面及背面。

(3) MUSIC DATA 必須要能利用 MOCKING BOARD 播送。

(4) 所有交來之磁碟必須在右上角貼上寫有下列資料的 LABEL：

假如擁有打字機，請將 PS 圖案硬本一併寄來。

姓名：　　電話：
身份証號碼：
參賽種類：PS/MCS （請註明A面或B面）
作品數目：　　個
作品名稱簡介：

(5) 所有交來之磁碟無論選用與否，將會在短時間內退回投稿者，如欲第一時間退回作品，請付回郵信封一個（要能容納磁碟）。

(6) MCS MUSIC DATA 投稿請用原稿紙將作品名稱寫下，如配有動畫請加簡略描述。

(7) 所有入選作品會安排在 COMPUSOFT 磁碟月刊刊出，得獎者我們將會用電話通知領取獎品。同時亦會在電腦時代刊出得獎者名單。

(8) 所有參賽作品以先到先取錄爲原則（如有相同題材，以先到先得）。

(9) 本比賽並不受時間限制，我們會一直舉辦下去。

(10) 參加次數不限。

(11) 所有入選作品皆有機會參加每六個月舉辦的「誰是總冠軍」大獎。冠軍將可獲得很豐富的獎品。

ATTENTION PLEASE !!

微電腦技術 Compu-Link

第三期經已出版

文章篇篇精彩
程式個個實用

APPLE

雙重磁碟系統格式化磁碟：
PRODOS 與DOS 3·3 並存一碟

可抄40軌磁碟的EDD III

40軌PASCAL 磁碟

磁碟修護程式：ZAP

CP/M

CP/M系統的剖析：
DDT及SAVE指令的使用

發揮TURBO PASCAL 的高效能

COBOL 80 使用機械語言的方法

組譯器ASM.COM 剖析

實用dBASE II 應用技巧：
加上保安系統設計的 MENU 系統

WORDSTAR 應用技術
多欄列印的技術探討

IBM

IBM XT 系統開關的設定

私人案頭助理系統

第三期十一月廿日經已出版
各大書報攤發售

■ 程式磁碟版本同時出版 ■

各期程式磁碟磁帶總目錄

1
1 突破傳統的鍵盤
2 撞球遊戲
3 勇闖電腦室
4 迷宮救美
5 將 APPLESOFT 變爲 FORTRAN
6 上、下、左、右移動文字畫面
7 測試工具— BOOLEAN 程式
8 騰空更多磁碟空間 一搬移HELLO
9 重現刪除之檔案
10 電腦時代月曆
11 COMPUTEXT 文字處理
12 時日相距計算程式
13 增強英文字體能力
14 幼童數學四則運算練習
15 過四關

2
1 貨櫃處理
2 沙盤幻方局
3 巧破密令
4 美術字繪劃器
5 小階字母處理
6 程式轉載器
7 彩色電視／監視器色彩測試
8 保留遊戲積分記錄
9 數學套件程式
10 磁碟修護工具
11 加設電腦時代月曆圖像一個

3
1 DOS 的自動同步追踪器
2 動量不滅定律模擬法
3 自製磁碟索引表封套
4 蘋果編曲器
5 幼童認數字初階
6 賽跑遊戲
7 步步追踪
8 DOS KILLER
9 圍棋遊戲
10 巴哈的古典樂章名作
11 畫轉螢幕畫面
12 雜誌目錄分類歸檔系統
13 神經病醫生一神經質健體操

4
(1) DOS 操作指令簡化器
(2) TEXT FILE 有些甚麼內容?
(3) 立體迷宮
(4) 英文串字練習
(5) 如何以機械語言來修補 DOS
(6) 電腦動畫製作原理。
(7) 豐富設計的學前兒童教育程式
(8) 應用數學套件程式(二)
(9) DOS WRITER
(10) 穿梭機哥倫比亞號機
(11) 找回失落了的 STEP 及 TRACE
(12) TRACER控制常式
(13) 將 RAM 卡作爲磁碟機使用
(14) 黑房計時器
(15) 太陽系模擬系統

5
1 私人約會備忘日曆
2 三度空間顯像系統
3 將 CP／M 檔案轉爲 DOS
4 蘋果萬花筒
5 高解像度文字產生器
6 修訂 DOS 操作指令簡化器
7 濃縮圖片儲存空間
8 幼童數學練習——數字咕
9 加數機大登場
10 備忘 MEMO紙
11 光速計數器
12 剩餘磁碟空 SECTOR數目
13 16進制輸入數字鍵盤
14 問題棋盤
15 神秘智號夾萬
16 隔山打牛
17 大空船降落任務

6
(1) 私人會計系統。
(2) APPLESOFT 擴展版本。
(3) GO }{ HELLO 的程式
(4) APPLESOFT 債錯系統。
(5) DOS 指令再修訂版本。
(6) 布倫時鐘。
(7) AMPERSAND 數基轉換器
(8) 完整索引錄。
(9) 砌積木。
(10) 神奇數字方塊。
(11) 亡命飛車。
(12) 迷宮搜秘。
(13) 太空怪侵略戰。
(14) 累加遊戲。
(15) 爲磁碟命名。
(16) 任意定義鍵盤。
(17) 捉蟲訓練問題。

7
(1) 會說話的蘋果
(2) 私人財政預算系統
(3) 私人收入會計系統
(4) 高解像字符繪劃器
(5) 圖像轉載器
(6) 螢幕文本編輯處理器
(7) 濃縮圖片程式修訂加強版本
(8) 蘋果杯大賽車
(9) 十六個難度的食鬼遊戲
(10) 高爾夫球遊戲。
(11) 空戰驚魂
(12) 美麗圖案設計家
(13) 更多色彩
(14) 列印的困擾
(15) 十一條空 SECTOR
(16) 捉蟲訓練
(17) 80字行咕例子程式

8
(一) APPLESOFT 黑編器
(二) 最簡單的拆帶程式
(三) 英文打字練習
(四) 編印單個LABEL
(五) 高解像頁處理工具箱
(六) 簡易快速的記憶移動方法
(七) 眞正要動腦筋的小學四則運算
(八) 七海霸王 (DEPTH CHARGE)
(九) 原子黑洞
(十) 蛇之噩夢
(十一) 收銀機
(十二) 改良的閃動訊息
(十三) 兩欄索引顯示
(十四) 自動三角函數曲綫繪劃系統
(十五) 任意鍵盤 (三個)
(十六) 每期捉蟲問題
(十七) 奇妙的 &

9
(1)磁碟檔案黃頁系統
(2)增強鍵盤輸入及螢幕輸出功能的系統
(3)將工具程式放入DOS佔用的三條磁軌上
(4)快速尋找及磁碟矩陣儲存與提取
(5)新DOS命令改換程式
(6)任意設定目錄索引顯示模式
(7)捉迷藏
(8)古堡之謎
(9)高空擲物
(10)有趣的排序
(11)捉虫問題及答案
(12)第八期捉虫問題答案
(13)印出高解像數目字

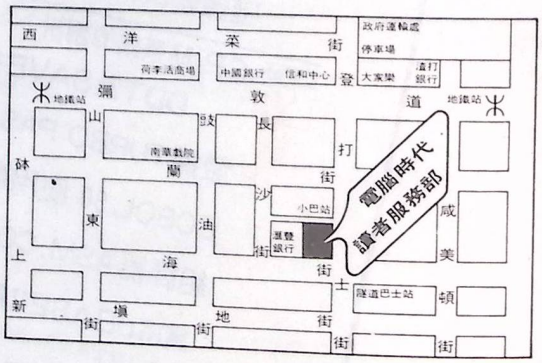
10
(1) 可隨意訂定格式之資料紀錄系統
(2) 太空戰士
(3) MARCO鍵盤
(4) 射鴨仔
(5) 奇妙的&——快速分類整理程式
(6) 40軌磁碟工具程式設計
(7) 變數表顯示
(8) 跳越加速器
(9) IQ方塊
(10) 快速檔案搜索
(11) 將DOS移上RAM咭
(12) 培養兒童的記憶能力
(13) 初中代數式運算
(14) 萬用數基轉換器
(15) 十六欄機械碼編印
(16) 捉蟲問題及答案

11
(1) 家庭購物備忘及存貨紀錄系統
(2) 可隨意訂定之資料紀錄系統 (下)
(3) 奇妙的&專欄——矩陣定值
(4) 40軌磁碟工具程式之(二) 單軌 INIT
(5) 兩個自動減省程式佔用之記憶空間應用程式
(6) 對我說
(7) 趣意盈盈的數樂棒
(8) 英文語句文法分析練習系統
(9) 一行程式示例 (共七個)
(10) 美化程式列表
(11) JACKIES HEAD
(12) 魔域迷宮
(13) 巨蟹求生記
(14) 自動日子記錄
(15) 空SEOCTOR圖表
(16) 捉虫訓練——FACTORIAL !

12
① 家庭開支及個人消費管理系統
② 電子郵箱
③ 40軌程式COPY工具
④ 諾曼第登陸戰 (模擬戰爭思考遊戲)
⑤ 工兵掘地雷
⑥ 新改良蛇之噩夢
⑦ 魔域迷宮 (APPLESOFT 版本)
⑧ 讓九個月大的嬰孩接觸電腦吧!
⑨ 有多項功能的AMPERSAND工具程式
⑩ 磁頭清潔器
⑪ 程式拯救者
⑫ TAB & XY
⑬ 符號逆輸型 (RPN) 計數機
⑭ 比RAM DISK 更好的RAM咭控制中心
⑮ 圖形建造師
⑯ 捉虫訓練——SINE曲線
⑰ 一行程式

精選程式磁碟

編號：CA00A
內容：遊戲及教育程式
遊戲程式
1、捉球遊戲 (第一期)
2、勇闖電腦室 (第一期)
3、迷宮救美 (第一期)
4、過四關 (第一期)
5、沙盤幻方局 (第二期)
6、貨櫃處理 (第二期)
7、賽跑遊戲 (第三期)
8、步步追踪 (第三期)
9、圍棋遊戲 (第三期)
10、立體迷宮 (第四期)
11、穿梭機哥倫比亞號 (第四期)
教育程式
12、增強英文匯能力 (第一期)
13、幼童數學四則運算 (第一期)
14、數學套件程式 (第二期)
15、動量不滅定律模擬法 (第三期)
16、幼童認數字初階 (第三期)
17、豐富設計的學前兒童教育程式 (第四期)
18、應用數學套件程式之二 (第四期)



編號：CA00B
內容：應用程式精選
1、電腦時代月曆 (第一期)
2、COMPUTEXT 文字處理 (第一期)
3、美術學繪劃器 (第二期)
4、彩色電視／監視器色彩測試 (第二期)
5、保留遊戲積分紀錄 (第三期)
6、自製磁碟索引封套 (第三期)
7、APPLE 編曲器 (第三期)
8、雜誌目錄分類歸檔系統 (第三期)
9、DOS WRITER (第四期)
10、電腦動畫製作原理 (第四期)
11、將 RAM咭作爲磁碟機使用 (第四期)

編號：CA00C
內容：私人會計系統
(1) 私人財政預算 (第七期)
(2) 私人收入紀錄 (第七期)
(3) 私人會計程式 (第六期)

編號：CA00D
內容：工具程式集
(1) APPLESOFT程式偵錯系統 (等六期)
(2) 將DOS 移上RAM 咭 (第十期)
(3) MACRO 鍵盤 (第十期)
(4) RAM 咭控制中心 (第十二期)
(5) 程式編輯器 (第十三期)

每期程式磁碟現仍售29.9元；
精選程式磁碟每張售價45元
(除CA00C 只售30元)。

歡迎補購各期程式磁碟或雜誌

各期程式磁碟總目錄

13

(1) 行政思考決策輔助系統——成功人士必備工具
(2) 全能標貼製作器
(3) 大力水手——POPEYE
(4) 火樹銀花賀聖誕——平安夜
(5) 低解像圖輔助繪劃器
(6) 連續文進化論的電腦模擬實踐
(7) 程式編輯器
(8) 磁碟檔案地圖
(9) 增加DOS的CATALOG暫停功能
(10) 新游標
(11) 40軌磁碟改動器
(12) 餅形圖
(13) 自己設信紙版頭
(14) 捉虫問題及答案
(15) 一行程式得樂作品

14A

(1) 高解像圖魔術師
(2) 以軟件來達成的音响合成器
(3) SCREEN SAVER
(4) 磁碟度身定造器
(5) DOS 的快速存取
(6) 一行程式得樂作品選
(7) HIBE：將程式隱藏
(8) 英文字義辨識練習
(9) 數學學習加、減運算
(10) 宇宙城保衛戰（麥法基專欄）
(11) 最後一個立體打井遊戲
(12) 月球一小步，人類一大步
(13) 音量控制
(14) 將APPLESOFT程式轉爲TEXT檔案來編輯
(15) 捉虫訓練——DOS指令的執行

14B

DOS 3.3 FORMAT (SIDE A)
(A) 令蘋果發言的最佳工具MOCKING BOARD SPEECH WRITER
(B) 本港郵費計算路徑系統
CP／M FORMAT (SIDE B)
(C) CP／M的密碼檔案
(D) 程式設計與FORTRAN所帶來的好處（雙精度計算例子）

15

(1) 活動電子訊息顯示板
(2) 借貸分期還款計算系統
(3) 麥法基遊戲專欄——DANGER RANGER
(4)「扁」臨香江
(5) 將高解像頁倒轉
(6) 機械碼輸入編輯器
(7) 程序分解擴展器
(8) 利用印字機印出多款字形
(9) 檔案密碼鎖
(10) 兩個新DOS 指令之一MREAD
(11) 電腦繪圖專欄——立體繪圖的原理示範程式
(12) 氫氧碳化學分子排列結構模擬
(13) 猫仔錢箱
(14) 3－6－9門智遊戲
(15) 超級越軌大賽
(16) 電腦音樂動畫——MONICA
(17) 一行程式入選作品
(18) 捉虫問題及答案

16

(1) 專業室內設計師
(2) 大忙人時間顯
(3) 學習迷宮產生的原理及自編迷宮遊戲
(4) 使用更方便的BOOT DRIVE 2 軟件
(5) LIST 大師
(6) CATALOG HELPER
(7) 40軌磁碟SECTOR 使用情況顯示
(8) 磁碟保護器——COPY VTOC 法
(9) 兩個新DOS 指令MREAD & MWRT（下）
(10) 爲你的程式配上背景音樂
(11) 成雙成對
(12) 吃不到的葡萄不是酸的 !!
(13) 動態／策略兼備的戰爭遊戲：BEACH HEAD
(14) 一行程式入選作品
(15) 捉虫問題及答案
(16) 組合線性方程式解答系統
(17) 太空侵略者封閉戰CLOSING－ALIEN

19

(1) 電腦時代文件櫃
(2) 辦公室下午茶備忘錄
(3) 蘋果超級500圈大賽車
(4) 五子連環（棋類遊戲百種之三）
(5) 高空救人
(6) 萬用輸入裝置
(7) 卡通人物專門店：小浣熊
(8) 配合卡通人物福印的月曆程式
(9) 時間教授
(10) 十進制換算表
(11) 利用MOCKING BOARD 作時鐘
(12) 認識一張適用於蘋果機的中文咭
(13) 自動檢定磁碟質素的DISK AID
(14) 三種美麗的展示高解像圖方法
(15) 直接讀取GAME FILE；活用RWTS
(16) 一行程式入選作品
(17) 第十九期捉虫問題

22

(1) PRINTSHOP 助手——印製私人名片及信紙
(2) 活用印字機
(3) 把機械語化成BASIC
(4) 從程式中學習編寫中文咭程式技巧——即唐詩遊戲
(5) 唱片櫃管理系統
(6) 電子收銀／計數機
(7) 16K WILDCARD
(8) 智力棋的考驗——MASTER MIND
(9) 德國橋牌
(10) 麥法基創作遊戲——立體死亡大賽車
(11) PRODOS 專欄——DATA・SET
(12) 磁碟壽命性能測試器
(13) 機械碼輸入錯誤核對工具
(14) 排序原理透視
(15) CRAZY PROTECTION
(16) LOAD RUNNER MAKER
(17) 80字行兩欄目錄檔案顯示

17

(1) 日文書信文字處理系統
(2) APPLESOFT 全域編輯器
(3) II PLUS 亦可擁有視窗顯示功能
(4) 仿效MOUSE PAINT設計的AP PAINT
(5) 商業貸款單
(6) 把損壞的磁軌鎖住
(7) 配合單軌INIT使用的INIT 40軌
(8) WILDCARD 解拆保護原理透視
(9) 兒童串字
(10) 英文聽力測驗
(11) 金字塔數字堆
(12) 麥法基遊戲專欄：TIME PILOT
(13) 武士忠魂
(14) MUSIC CONSTRUCTION SET 背景音樂問題
(15) GRAPHIC SOLUTION 動畫DEMO
(16) VISICALC 篇——DEMO
(17) 一行程式優選作品
(18) 第十七期捉虫訓練

20

(1) 快速DATA 處理程式
(2) 印字機功能大揭秘
(3) 中文咭應用專欄——生肖年份檢索
(4) 電子線路設計工具套件
(5) 另一個功能最強程式——POWER KEY
(6) 麥法基遊戲專欄：DEATH RACE
(7) 棋類遊戲百種之四——四位一體
(8) 暗渠截擊戰
(9) 麥法基專欄：創造一個移動的立體背景DEMO
(10) 機械語言專欄：程式執行速度的探討
(11) HELLO程式漫談／HELLO BREAKER
(12) 觀看文字檔案
(13) 輔助程式編輯器
(14) 從足球運動中學習物理現象
(15) 看圖學英文生字
(16) 從文字幕到高解像頁
(17) 改良TRACK & FIELD

23

(1) 濃縮及還原高解像圖原理剖析
(2) 統計及分析各期結果之六合彩分析器
(3) 中文大字海報編印器
(4) 家庭資產管理系統
(5) 郵資路徑系統修訂版
(6) LAST ONE 最後一字
(7) 按足簡京規例的廿一點紙牌遊戲
(8) 化學實驗模擬
(9) 35／40磁軌快速COPY 程式
(10) 專爲PRINTSHOP 而設的製圖工具
(11) 螢幕顯示方法製造器
(12) 修訂版本之CRAZY PROTECTION
(13) 配合私人約會備忘日曆使用的MEMO 顯示器
(14) 內置DOS 的FID
(15) 光速數行器

18

(1) 配合PRINT SHOP 使用賀咭信封套印製
(2) 圖像／文字顯示選擇器
(3) 股票投資分析系統
(4) 麥法基遊戲專欄：RAIN SHOWER
(5) 書信文件編打系統
(6) 方天塔爭霸戰
(7) 步步爲營
(8) VISICALC 篇：函數@LOOK UP 的使用技巧
(9) 卡通人物專門店：藍精靈
(10) 磁碟LOCKER
(11) VTOC 編輯器
(12) 利用左右箭咀鍵選擇的CATALOG HELLO
(13) 從六合彩到編寫程式技巧
(14) 生動活潑認數目遊戲
(15) 使用BASIC 在高解像畫面玩魔術
(16) 機械碼列印對照表
(17) 一行程式入選作品選
(18) 第十八期捉虫問題

21

(1) 模擬印字機BUFFER 咭
(2) 萬用統計圖表製作器
(3) 核子嬌娥
(4) 棋類遊戲之五——海戰棋
(5) 麥法基創作遊戲——STAR WAR
(6) PRODOS 專欄——PROCAT
(7) 機械語言應用——SLIDE SHOW系統
(8) 有七項功能的DOS 設計器
(9) 程式執行監察器
(10) 文字幕旋轉器
(11) 函數擴展及因子分解
(12) 智力棋——TEST MIND
(13) 逐漸擴大方孔式畫面顯示

電腦時代合訂本新增內容程式磁碟

電腦時代精裝合訂本（1－12期）
新增內容程式磁碟集：
(1) 將低解像圖或文字頁存入RAM咭
(2) 將高解像圖頁存入RAM咭
(3) 磁碟操作系統專家——RWTS 原程式
每張只售15元
歡迎到讀者服務部購買或利用郵購方式。
郵購辦法請參閱郵購需知欄。

歡迎補購

電腦時代 分類廣告

「電腦時代」由第由21期開始，增設分類小廣告欄歡迎各界賜稿惠刊。

每段小廣告：**大字10個，小字50個，收費30元。**

＊來稿請繕寫清楚，免有校誤。
＊凡涉及法律性廣告，請携同身份證。
＊每月十號前截稿；來稿可直接交來「電腦時代」讀者服務部，或以劃線支票（抬頭填寫「電腦時代出版社」）郵寄九龍中央郵政信箱71193號。
＊廣告費請先惠付（超過三期九折優待；超過六期者，更獲八折優待），刊字如超過一段照兩段計。
＊每一廣告客刊，最多只能刊登三段。

_____先生/小姐/女士/公司

茲附上港幣 _____元（現金/支票№ _____）請將附上之廣告代爲刊登；由「電腦時代」_____期開始，至_____期止，合共_____則。此致

電腦時代出版社

刊戶：_____

地址：_____

_____

電話：_____ 日期：_____

電腦時代程式磁碟

每張只售HK$29.90

電腦時代每期都刊登多個精彩的程式，其中必然有你喜歡的。當你讀完文章興緻勃勃的坐下來欲進一步深入了解程式的精彩處時，看見那大堆冗長的列表，就甚麼興趣也沒有了！那麼，你爲何不購買一張「電腦時代程式磁碟」呢！

每期出版的電腦時代所登載的程式，均全部紀錄入一張高品質的磁碟上，並隨雜誌同時出版。爲省卻打入程式的麻煩，徒然浪費你寶貴的時間，免除了花精神捉錯打字的痛苦，今日就請就近的特約經銷處或電腦時代讀者服務部/COMPUHOUSE購備一張。

程式磁碟每張只售港幣 29.90 元。

第24期程式磁碟內容

*All Programs are in Apple version

(1) Ⅱ十上的雙倍高密度解像效果
(2) 改良版中文大字海報——大小字隨意
(3) 趨吉避凶的生理周期分析預報系統
(4) 被困在迷宮的小猩猩
(5) 棋類遊戲百種之八：戰略跳棋(TACTICS)
(6) 機械語言應用：起始地址搬移器
(7) 方向概念學習——東南西北
(8) 控制字符檢察器
(9) 程式保留字檢查系統
(10) 有預報功能的新INIT指令
(11) 專爲拷貝BROTHERBUND 軟件而設的POWER COPY
(12) 拆LOCKSMITH 5.0 16 SECTOR FAST COPY 爲FILE
(13) 加上CLOCK 咭應用的私人約會備忘

爲方便讀者購買程式磁碟，特新增香港電視服務站爲我們特約經銷處;現前往各大地鐵及九廣鐵路電視服務站均有代售。

香港電視服務站

太古城服務站
太古城中心（太古城道）

地鐵站

荃灣服務站	寶富服務站
葵芳服務站	九龍塘服務站
美孚服務站	石硤尾服務站
荔枝角服務站	旺角服務站
長沙灣服務站	油麗地服務站
觀塘服務站	佐敦服務站
九龍灣服務站	尖沙咀服務站
彩虹服務站	金鐘服務站
黃大仙服務站	中環服務站

九廣鐵路車站

九龍塘服務站	大圍服務站
旺角服務站	沙田服務站

中環

百老滙攝影器材有限公司（第五店）
域多利皇后街3號（中環街市側）
電話：5－235513

亞富影音
士丹利街 28A（陸羽茶樓側）
電話：5－240723

三聯書店讀者服務中心
香港中環域多利皇后街九號（中環街市側）
電話：5~250102

銅鑼灣

百老滙攝影器材有限公司（第十四店）
軒尼斯道529號地下（三越百貨公司對面）
電話：5－770181

太古城

百老滙攝影器材有限公司（第七店）
太古城中心第232店
電話：5－672006

西環

創業雜誌中心
西環屈地街22號地下　明愛中心對面
電話：5-478338

旺角

百老滙攝影器材有限公司　（第1,3,9,10店）
彌敦道758號在下（凱聲戲院側）
電話：3－816931
彌敦道700號旺角中心第二期地下
電話：3－943827
豉油街26號地下（近商務書局）
電話：3－884331
亞皆老街 60B 地下（世界書局對面）
電話：3－956367

深水步

百老滙攝影器材有限公司（第十三店）
青山道148號地下（華聲戲院對面）
電話：3－7295938

尖沙咀

智源書局
金巴利道 27－33號永利大厦2字樓A座
電話：3 － 678482　3－678414

荔枝角

百老滙攝影器材有限公司（第二店）
美孚新邨第四期百老滙街 79號（影都戲院側）
電話：3－7424923

觀塘

百老滙攝影器材有限公司（第四、十二店）
物華街3號地下（皇上皇側）
電話：3－419513
同仁街35號地下（大家樂側）
電話：3－449940

荃灣

百老滙攝影器材有限公司（第十一店）
沙咀道 230A（戴麟趾夫人健康院對面）
電話：0-4938926

北角

資訊電腦中心
北角英皇道121號七海商業中心134室
電話：5-708301

高登/黃金商場特約經銷處：
深水步福華街黃金商場地庫21號

教育專業人員協會（北角會所）

電腦時代讀者服務部/ CompuHouse

地址：旺角上海街438號同珍商業中心11字樓1104室
（上海街／登打士街交界）　電話：3-312007

補購各期程式磁碟可前往讀者服務部／COMPU HOUSE

Apple is the trademark of Apple Computer Inc.

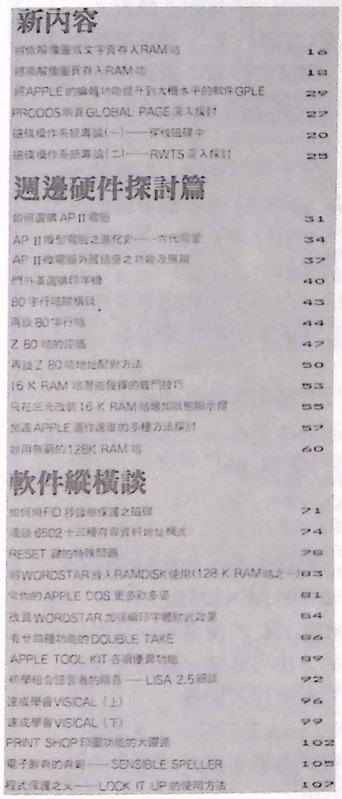
電腦時代精裝珍藏本 (1-12期合訂本)

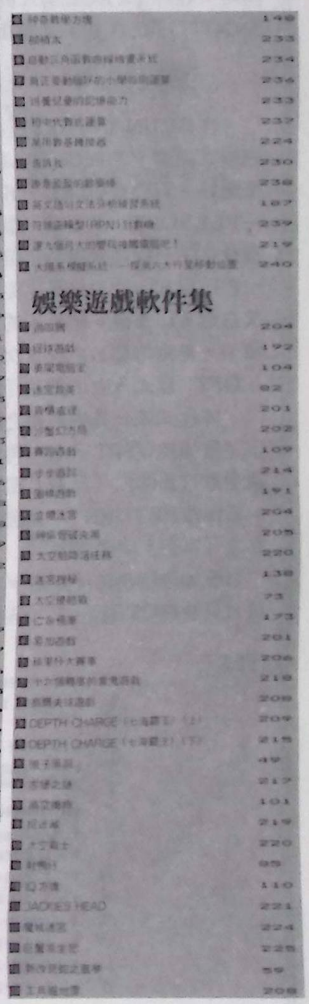
隨書附送價值超過 150 元的購買程式磁碟優待券

精裝珍藏本，顧名思義，是將過去出版的電腦時代第 1 至12期所有的精彩內容滙集成一巨册，其中包括超過一百個的軟件程式使用說明，數十篇有關各類週邊硬件深入解剖報告、熱門軟件評介及用法文章。是所有擁有 APPLE或 APPLE兼容電腦的用家必須珍藏的巨獻。

可能你已擁有第一至十二期，但這册精裝珍藏本依然有你值得擁有的價值，因爲我們在這册合訂本中加插了多篇從未刊登過的文章及程式。而且我們更隨書附送價值超過150元的購買程式磁碟優待券。

電腦時代精裝珍藏本全書厚度超過200餘頁，採用穩固的膠背釘裝。每册僅售50元。





特約經銷處：

- 電腦時代讀者服務部
- 中環三聯書局
- 教育專業人員協會（北角會址）
- 深水埗黃金商場地庫 21

程序設計員的助手

程式保留字檢察系統

今期我們爲喜愛編寫程序的朋友介紹一個專捉字蚤的工具程式，相信對無論是自己創作程序，又或者是研究欣賞別人的作品，也會有很大的幫助。

這系統是以MERLIN ASSEMBLER 寫成然後組譯成機械碼執行（見列表1），不會騷擾APPLESOFT 的記憶體，但需要最少64K 的記憶才可以操作。它能令指定的AP－PLESOFT 保留字或指令在程式列表（LIST）時，反白顯示。系統只需用簡單的指令就可以，而在執行時亦有適當的螢幕指示。

相信閣下不難想像一個反白的指令在其他以正常狀態顯示的字叢中，是十分突出的，所以，當我們要找尋某些指令，以便進行修改的時候，這系統便大派用場了。比如要改FOR－NEXT 廻路的參數，HTAB 和VTAB 的位置等。同時，在你TRACE別人的程式時，亦可以把所有在程式中的GOTO「反白」了，讀起來便會更方便和準確精細。

系統操作

在BRUN INVERSE KEYWORD（本程式在同期出版的程式磁碟上的FILE NAME）後，系統立刻告訴你它已經準備好，只要按CTRL－K便可以輸入你想「突出」的AP－PLESOFT 保留字（指令），若你的輸入合乎格式，游標會在下一行如常顯示，但當你LIST 記憶內的APPLESO－FT 程式，便會看見你剛才輸入的「保留字」已經全部以反白形式印出來。你輸入的「保留字」會一直維持「反白」顯示，無論你LIST 多少次，或者LOAD 另一個APPLES－OFT 程式入去，那個「保留字」都會以反白形式顯示。

本程式唯一美中不足之處是每次只能「反白」一個保留字，當你按CTRL－K 再輸入另一個保留字，新的反白字便會取代舊的以「反白」顯示，若你按CTRL－K 後不答覆，直接按RETURN ，系統便打出舊有的保留字，並維持舊有之「反白」字體顯示。

假如你想消除「反白」字的顯示，回復正當字體顯示的話，只要按CTRL－J ，系統便打出 "OFF" 字樣，表示系統暫時解除。但那「字」仍留在記憶之中，要令該「字」又再用反白顯示，可按CTRL－K 再直接打RETURN， 或者按RESET ，便再把系統連結起來。

若閣下想以印字機列出程式，宜先以CTRL－J 脫離系統，因爲那些以「反白」顯示的字體不單祇不能印出來，對於某些印字機，可能會遭遇不可控制之「亂印」情況。

若你想永遠截斷系統，可按CTRL－Z ，則系統便無法再進入了，因爲RESET 指標，HIMEN，輸入掛鈎都重置到未BRUN系統前的數值，臨別前系統會打出「ZAP ！」 ，表示玩完了。

最後要補充的是輸入「保留字」式「指令」的格式。

有括號跟着的指令如ASC，CHR $，等，是不用輸入其括號部份的。有參數的指令如HTAB，VTAB，HLIN，VLIN 等，其數字部份是不被接納的。

一些數學和特殊符號如大於＞，小於＜相等＝，乘＊除／，加＋，減－等，和APPLESOFT 的特殊指令＆，在本系統中皆被接納爲正確指令。

但注意用來分隔指令的 "：" 是不被接納的輸入。

輸入時須注意指令後面不可以有空白，空白是不被接納的，所以，指令入完便應直接打RETURN。

輸入程式

本程式系統可見於列表1，列表1是一個用MERLIN ASSEMBLER 寫成的程式，懂得用ASSEMBLER 的讀者可一一鍵入。未曾學過ASSEMBLER 的讀者可依照列表左邊的機械碼輸入。輸入機械碼的方法可參閱「新讀者需知」。輸入完機械碼列表後可用下列指令存入磁碟：

BSAVE INVERSE KEYWORD，A$9353，L$2AB

或者大家可在同期出版的程式磁碟上找到本程式的OBJ・FILE 及SOURCE CODE。

最後要提醒大家一樣就是要執行本程式，必須要有64K 記憶系統或以上。 ■

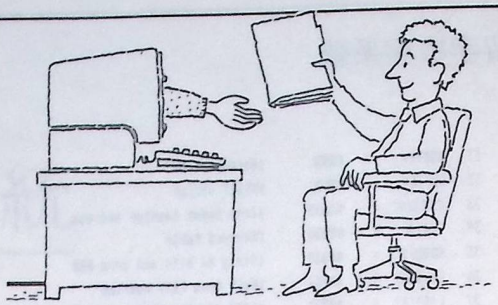
列表一

```
1   ************************
2   *  INVERSE KEYWORDS  *
3   *                    *
4   * COPYRIGHT (C) 1985 *
5   * BY   COMPUHOUSE    *
6   *                    *
7   ************************
8   *  Requirements:
9   *   1. HIMEM:38400 ($9600) or more
10  *   2. Applesoft on motherboard
11  *   3. 64K RAM
12  *  Commands:
13  *   1. CTL-K (program on; get keyword)
14  *   2. CTL-J (program off)
15  *   3. CTL-Z (disconnect program)
16  * Merlin Assembler
17
18  PTR     =   $06     ;Pointer for print subroutine
19  CH      =   $24     ;Horizontal cursor position
20  BASL    =   $28     ;Left margin of current line
21  PROMPT  =   $33     ;Prompt character
22  KSWL    =   $38     ;Input hook
23  A1L     =   $3C     ;MOVE parm: Start
24  A2L     =   $3E     ;           End
25  A4L     =   $42     ;           Destination
26  FRETOP  =   $6F     ;Bottom of strings
27  MEMSIZ  =   $73     ;HIMEM
28  FORPNT  =   $85     ;Store index to TKNTBL
29  DSCTMP  =   $9D     ;Pointer to start of TKNTBL
30  BUF     =   $200    ;Input buffer
```

程式保留字檢察系統

```
                        31  DOSWRM  =   $3D0      ;Warmstart DOS
                        32  SOFTEV  =   $3F2      ;RESET vector
                        33  KSWTRUE =   $AA55     ;True input handler address
                        34  TKNTBL  =   $D0D0     ;Keyword table
                        35  GDBUFS  =   $9539     ;Strip hi bits and zero EOB
                        36  LISTOUT =   $D736     ;Exit from LIST routine
                        37  LISTNI  =   $D739     ;LIST reentry (token found)
                        38  LISTIN2 =   $D762     ;LIST reentry (token not found)
                        39  OUTSP   =   $DB57     ;Output space
                        40  BASICCLD =  $E000     ;Coldstart Basic ($4C=Apsoft)
                        41  SETPWRC =   $FB6F     ;Set power-up byte
                        42  BS      =   $FC10     ;Backspace
                        43  CLREOL  =   $FC9C     ;Clear to end of line
                        44  KEYIN   =   $FD1B     ;Input character (bypass KSW)
                        45  RDCHAR  =   $FD35     ;Input character (thru KSW)
                        46  CROUT   =   $FD8E     ;Output CR
                        47  COUT    =   $FDED     ;Output character
                        48  MOVE    =   $FE2C     ;Move memory (Y=0)
                        49  BELL    =   $FF3A     ;Beep!
                        50
                        51          ORG  $9353
                        52
                        53  *==============
                        54  * Installation:
                        55  *==============
                        56  * Check HIMEM:
                        57
9353: A5 74             58          LDA  MEMSIZ+1
9355: C9 96             59          CMP  #$96
9357: B0 09             60          BCS  CKROM      ;HIMEM >= $9600
9359: A0 62             61          LDY  #TXHMEM    ;Print "HIMEM BELOW $9600"
935B: A9 95             62          LDA  #>TXHMEM
935D: 20 51 95          63          JSR  PRINT
9360: F0 36             64          BEQ  BASIC1     ;Always
                        65
                        66  * Check Applesoft ROM on motherboard:
                        67
9362: 8D 82 C0          68  CKROM   STA  $C082     ;Motherboard ROM read
9365: AD 00 E0          69          LDA  BASICCLD
9368: C9 4C             70          CMP  #$4C
936A: F0 09             71          BEQ  CK64K      ;Apsoft on motherboard
936C: A0 7A             72          LDY  #TXROM     ;Print "APPLESOFT NOT
936E: A9 95             73          LDA  #>TXROM    ; ON MOTHERBOARD!"
9370: 20 51 95          74          JSR  PRINT
9373: F0 23             75          BEQ  BASIC1     ;Always
                        76
                        77  * Check 64K configuration:
                        78
9375: AD 83 C0          79  CK64K   LDA  $C083     ;Bank 2 R/W
9378: AD 83 C0          80          LDA  $C083
937B: A9 4F             81          LDA  #$4F       ;Any nonzero value will do
937D: 8D 00 D0          82          STA  $D000
9380: CD 00 D0          83          CMP  $D000
9383: D0 09             84          BNE  NO64K      ;Write failed. Not 64K
9385: 0A                85          ASL             ;Try once again (to avoid chance
9386: 8D 00 D0          86          STA  $D000      ; duplication of byte)
9389: CD 00 D0          87          CMP  $D000
938C: F0 0D             88          BEQ  SAVDATA    ;64K present. Proceed
938E: 8D 82 C0          89  NO64K   STA  $C082     ;Motherboard ROM read, bank 2 WP
9391: A0 9E             90          LDY  #TX64K     ;Print "64K RAM NOT PRESENT"
9393: A9 95             91          LDA  #>TX64K
9395: 20 51 95          92          JSR  PRINT
9398: 4C D0 03          93  BASIC1  JMP  DOSWRM
                        94
                        95  * Save current HIMEM, input handler and RESET vector:
                        96
939B: A5 73             97  SAVDATA LDA  MEMSIZ
939D: 8D EB 95          98          STA  HMEMSAV
93A0: A5 74             99          LDA  MEMSIZ+1
93A2: 8D EC 95         100          STA  HMEMSAV+1
93A5: AD 55 AA         101          LDA  KSWTRUE
93A8: 8D ED 95         102          STA  INSAV
93AB: AD 56 AA         103          LDA  KSWTRUE+1
93AE: 8D EE 95         104          STA  INSAV+1
93B1: AD F2 03         105          LDA  SOFTEV
93B4: 8D EF 95         106          STA  RSETSAV
93B7: AD F3 03         107          LDA  SOFTEV+1
93BA: 8D F0 95         108          STA  RSETSAV+1
                       109
                       110  * Move ROM to bank-switched memory (bank 2):
                       111
93BD: 8D 81 C0         112          STA  $C081      ;ROM read, bank 2 write
93C0: 8D 81 C0         113          STA  $C081
93C3: A9 D0            114          LDA  #$D0
93C5: 85 3D            115          STA  A1L+1
93C7: 85 43            116          STA  A4L+1
93C9: A0 FF            117          LDY  #$FF
93CB: 84 3E            118          STY  A2L
93CD: 84 3F            119          STY  A2L+1
93CF: C8               120          INY             ;(Y)=0
93D0: 84 3C            121          STY  A1L
93D2: 84 42            122          STY  A4L
93D4: 20 2C FE         123          JSR  MOVE
                       124
                       125  * Modify LIST routine:
                       126
93D7: 8D 83 C0         127          STA  $C083      ;Bank 2 R/W
93DA: 8D 83 C0         128          STA  $C083
93DD: A9 4C            129          LDA  #$4C       ;Place "JMP PRTKN" into LIST
93DF: 8D 36 D7         130          STA  LISTOUT    ; routine on bank 2
93E2: A9 37            131          LDA  #PRTKN
93E4: 8D 37 D7         132          STA  LISTOUT+1
93E7: A9 95            133          LDA  #>PRTKN
93E9: 8D 38 D7         134          STA  LISTOUT+2
                       135
                       136  * Establish RESET handler:
                       137
93EC: A9 00            138          LDA  #RESET     ;Point page 3 reset vector at
93EE: 8D F2 03         139          STA  SOFTEV     ; program reset handler
93F1: A9 94            140          LDA  #>RESET
93F3: 8D F3 03         141          STA  SOFTEV+1
93F6: 20 6F FB         142          JSR  SETPWRC
                       143
                       144  * Print "INSTALLED":
                       145
93F9: A0 B8            146          LDY  #TXINST
93FB: A9 95            147          LDA  #>TXINST
93FD: 20 51 95         148          JSR  PRINT
                       149
                       150  * Fall into RESET:
                       151
                       152  *==============
                       153  * RESET handler:
                       154  *==============
9400: 8D 83 C0         155  RESET   STA  $C083     ;Bank 2 read
                       156
                       157  * Protect program above HIMEM and set bottom of strings:
                       158
9403: A9 00            159          LDA  #RESET
9405: 85 73            160          STA  MEMSIZ
9407: 85 6F            161          STA  FRETOP
9409: A9 94            162          LDA  #>RESET
940B: 85 74            163          STA  MEMSIZ+1
940D: 85 70            164          STA  FRETOP+1
                       165
                       166  * Reset input hook and notify DOS:
                       167
940F: A9 1A            168          LDA  #INPUT
9411: 85 38            169          STA  KSWL
9413: A9 94            170          LDA  #>INPUT
9415: 85 39            171          STA  KSWL+1
9417: 4C D0 03         172          JMP  DOSWRM
                       173
                       174  *==============
                       175  * Input handler:
```

程式保留字檢察系統



```
                        176 *===============
941A: 20 1B FD          177 INPUT   JSR KEYIN
                        178
                        179 * Command valid in immediate Basic mode, with
                        180 * cursor just to right of Applesoft prompt:
                        181
941D: 48                182         PHA
941E: A5 33             183         LDA PROMPT
9420: C9 DD             184         CMP #"]"
9422: D0 14             185         BNE RTSI        ;Invalid prompt character
9424: A5 24             186         LDA CH
9426: C9 01             187         CMP #1
9428: D0 0E             188         BNE RTSI        ;Invalid position
942A: 68                189         PLA
                        190
                        191 * Check for valid command:
                        192
942B: C9 8B             193         CMP #"K"-$40    ;CTL-K
942D: F0 44             194         BEQ KCMD
942F: C9 8A             195         CMP #"J"-$40    ;CTL-J
9431: F0 33             196         BEQ JCMD
9433: C9 9A             197         CMP #"Z"-$40    ;CTL-Z
9435: F0 03             198         BEQ ZCMD
9437: 48                199         PHA             ;Invalid command. Kill next PLA
9438: 68                200 RTSI    PLA
9439: 60                201         RTS
                        202 *=======================
                        203 * CTL=Z command handler:
                        204 *=======================
                        205 * Restore HIMEM, input and RESET vectors and enable ROM:
                        206
943A: AD EB 95          207 ZCMD    LDA HMEMSAV
943D: 85 73             208         STA MEMSIZ
943F: AD EC 95          209         LDA HMEMSAV+1
9442: 85 74             210         STA MEMSIZ+1
9444: AD ED 95          211         LDA INSAV
9447: 85 38             212         STA KSWL
9449: AD EE 95          213         LDA INSAV+1
944C: 85 39             214         STA KSWL+1
944E: AD EF 95          215         LDA RSETSAV
9451: 8D F2 03          216         STA SOFTEV
9454: AD F0 95          217         LDA RSETSAV+1
9457: 8D F3 03          218         STA SOFTEV+1
945A: 20 6F FB          219         JSR SETPWRC
                        220
                        221 * Print "ZAP":
                        222
945D: A0 C5             223         LDY #TXZAP
945F: A9 95             224         LDA #>TXZAP
9461: 20 51 95          225         JSR PRINT
9464: F0 07             226         BEQ J1          ;Always
                        227 *=======================
                        228 * CTL-J command handler:
                        229 *=======================
                        230 * Print "OFF":
                        231
9466: A0 CC             232 JCMD    LDY #TXOFF
9468: A9 95             233         LDA #>TXOFF
946A: 20 51 95          234         JSR PRINT
                        235
                        236 * Disable bank-switched memory and return to Basic:
                        237
946D: 8D 82 C0          238 J1      STA $C082       ;ROM read, bank 2 WP
9470: 4C B0 03          239         JMP DOSWRM      ;Back to Basic
                        240 *=======================
                        241 * CTL-K command handler:
                        242 *=======================
9473: 8D 83 C0          243 KCMD    STA $C083       ;Bank 2 R/W
9476: 8D 83 C0          244         STA $C083
                        245
                        246 * Solicit name of keyword to highlight:
                        247
9479: A0 D1             248         LDY #TXINP      ;Print "KEYWORD=)"
947B: A9 95             249         LDA #>TXINP
947D: 20 51 95          250         JSR PRINT
                        251
                        252 * Collect input:
                        253
9480: A2 00             254         LDX #0          ;Character index
9482: 20 35 FD          255 GETKEYW JSR RDCHAR      ;Get character
9485: C9 83             256         CMP #"C"-$40    ;CTL-C aborts
9487: F0 3C             257         BEQ GOBSIC      ; to Basic
9489: C9 8D             258         CMP #"M"-$40    ;CTL-M is CR
948B: F0 2F             259         BEQ RTN
948D: C9 88             260         CMP #"H"-$40    ;CTL-H is backspace
948F: F0 22             261         BEQ BCK
9491: C9 95             262         CMP #"U"-$40    ;CTL-U is copyover
9493: D0 14             263         BNE CTLTEST
9495: A4 24             264         LDY CH
9497: B1 28             265         LDA (BASL),Y    ;Copyover character at cursor
9499: 9D 00 02          266 ADDIN   STA BUF,X       ;Stuff into BUF
949C: 20 ED FD          267         JSR COUT        ;Print the character
949F: E8                268         INX
94A0: E0 09             269         CPX #9
94A2: 90 DE             270         BCC GETKEYW     ;9 or fewer characters allowed
94A4: A9 00             271         LDA #0          ;More than 9 characters
94A6: 4C 2D 95          272         JMP BADKEY      ; triggers error message
94A9: C9 A0             273 CTLTEST CMP #$A0
94AB: B0 EC             274         BCS ADDIN       ;No CTL character. Proceed
94AD: 20 3A FF          275 ERRINP  JSR BELL        ;CTL character. Ring bell
94B0: 4C 82 94          276         JMP GETKEYW     ; and get more input
94B3: 8A                277 BCK     TXA             ;Disable backspace if it is
94B4: F0 F7             278         BEQ ERRINP      ; first input character
94B6: 20 10 FC          279         JSR BS          ;Okay to back up
94B9: CA                280         DEX             ;Adjust index
94BA: 10 C6             281         BPL GETKEYW     ;Always
94BC: 9D 00 02          282 RTN     STA BUF,X       ;Final character (CR)
                        283
                        284 * Check whether input was made:
                        285
94BF: 8A                286         TXA             ;If (X)=0 then no input
94C0: D0 0E             287         BNE TKNIZE      ;Input present. Proceed
94C2: AD F1 95          288         LDA TOKEN       ;No input. Check default keyword
94C5: F0 4B             289 GOBSIC  BEQ BSIC        ;No default keyword
94C7: A0 F3             290         LDY #KEYWORD    ;Print default keyword
94C9: A9 95             291         LDA #>KEYWORD
94CB: 20 51 95          292 PRBSIC  JSR PRINT
94CE: F0 42             293         BEQ BSIC        ;Always
94D0: 20 9C FC          294 TKNIZE  JSR CLREOL      ;Clean up line
94D3: 20 39 B5          295         JSR GDBUFS      ;Zero EOB and strip hi bits
                        296
                        297 * Tokenize and check validity of keyword:
                        298
94D6: A9 D0             299         LDA #TKNTBL     ;Set keyword table
94D8: 85 9D             300         STA DSCTMP      ; pointer at $CFD0
94DA: A9 CF             301         LDA #>TKNTBL-$100
94DC: 85 9E             302         STA DSCTMP+1
94DE: A0 00             303         LDY #0          ;(Y) indexes table characters
94E0: 8C F2 95          304         STY TKNINDX     ;TKNINDX indexes keywords
94E3: 88                305         DEY
94E4: A2 FF             306         LDX #$FF
94E6: C8                307 KEYCHR  INY
94E7: D0 02             308         BNE BUFCHR
94E9: E6 9E             309         INC DSCTMP+1    ;(DSCTMP),Y=$D0D0 on 1st pass
94EB: E8                310 BUFCHR  INX
```

程式保留字檢察系統

```
94EC: BD 00 02  311          LDA  BUF,X      ;Get input character
94EF: 48        312          PHA
94F0: 29 3F     313          AND  #$3F       ;Convert to inverse and
94F2: 9D F3 95  314          STA  KEYWORD,X  ; store in buffer
94F5: 68        315          PLA
94F6: 38        316          SEC
94F7: F1 9D     317          SBC  (DSCTMP),Y ;Check match with table character
94F9: F0 EB     318          BEQ  KEYCHR     ;Nonfinal character match
94FB: C9 80     319          CMP  #$80
94FD: D0 19     320          BNE  NXKEY      ;No match. Try next keyword
94FF: 0D F2 95  321          ORA  TKNINDX    ;Final match. TKNINDX plus $80
9502: 8D F1 95  322          STA  TOKEN      ; equals value of token
9505: E8        323          INX
9506: BD 00 02  324          LDA  BUF,X      ;If no other character has been
9509: F0 04     325          BEQ  MATCH      ; input then keyword valid
950B: A9 00     326          LDA  #0         ;If another char was input then
950D: F0 1E     327          BEQ  BADKEY     ; keyword invalid (e.g.HOMER)
950F: 9D F3 95  328 MATCH    STA  KEYWORD,X  ;Zero end of keyword buffer
9512: 20 8E FD  329 BSIC     JSR  CROUT
9515: 4C 00 94  330          JMP  RESET      ;Back to Basic
9518: EE F2 95  331 NXKEY    INC  TKNINDX    ;Bump keyword index
951B: B1 9D     332 NK1      LDA  (DSCTMP),Y ;Get table character
951D: C8        333          INY
951E: D0 02     334          BNE  NK2
9520: E6 9E     335          INC  DSCTMP+1
9522: 0A        336 NK2      ASL             ;Bit 7 into carry
9523: 90 F6     337          BCC  NK1        ;Nonfinal character
9525: B1 9D     338          LDA  (DSCTMP),Y ;Get 1st char in next keyword
9527: F0 04     339          BEQ  BADKEY     ;End of table. No such keyword
9529: A2 FF     340          LDX  #$FF
952B: D0 BE     341          BNE  BUFCHR     ;Test next keyword in table
952D: AA        342 BADKEY   TAX
952E: 8E F1 95  343          STX  TOKEN      ;Zero TOKEN=invalid keyword
9531: A0 DC     344          LDY  #TXERR
9533: A9 95     345          LDA  #/TXERR
9535: D0 94     346          BNE  PRBSIC     ;Always
                347 *=========================================
                348 * Token handler - LIST routine routed here:
                349 *=========================================
                350 * Check the token being processed:
                351
9537: CD F1 95  352 PRTKN    CMP  TOKEN
953A: D0 0F     353          BNE  NOTFND
                354
                355 * It's the one we want. Print it inversely:
                356
953C: 20 57 DB  357          JSR  OUTSP
953F: 84 05     358          STY  FORPNT     ;Save (Y)
9541: A0 F3     359          LDY  #KEYWORD
9543: A9 95     360          LDA  #/KEYWORD
9545: 20 51 95  361          JSR  PRINT
9548: 4C 62 D7  362          JMP  LISTIN2
                363
                364 * It's not the token we want. Reenter LIST routine:
                365
954B: 38        366 NOTFND   SEC
954C: E9 7F     367          SBC  #$7F
954E: 4C 39 D7  368          JMP  LISTIN1
                369 *==================
                370 * Print subroutine:
                371 *==================
9551: 84 06     372 PRINT    STY  PTR
9553: 85 07     373          STA  PTR+1
9555: A0 00     374          LDY  #0
9557: B1 06     375 PRT1     LDA  (PTR),Y
9559: F0 06     376          BEQ  RTS2
955B: 20 ED FD  377          JSR  COUT
955E: C8        378          INY
955F: D0 F6     379          BNE  PRT1       ;Always
9561: 60        380 RTS2     RTS
                381 *==================
                382 * Text of messages:
                383 *==================
9562: 87 8D 8D  384 TXHMEM   HEX  878D8D
9565: 20 08 09
9568: 0D 05 0D
956B: 20 02 05
956E: 0C 0F 17
9571: 20 24 39
9574: 36 30 30
9577: 20        385          INV  ' HIMEM BELOW $9600 '
9578: 8D 00     386          HEX  8D00
957A: 87 8D 8D  387 TXROM    HEX  878D8D
957D: 20 01 10
9580: 10 0C 05
9583: 13 0F 06
9586: 14 20 0E
9589: 0F 14 20
958C: 0F 0E 20
958F: 0D 0F 14
9592: 08 05 12
9595: 02 0F 01
9598: 12 04 21
959B: 20        388          INV  ' APPLESOFT NOT ON MOTHERBOARD! '
959C: 8D 00     389          HEX  8D00
959E: 87 8D 8D  390 TX64K    HEX  878D8D
95A1: 20 36 34
95A4: 0B 20 12
95A7: 01 0D 20
95AA: 0E 0F 14
95AD: 20 10 12
95B0: 05 13 05
95B3: 0E 14 20  391          INV  ' 64K RAM NOT PRESENT '
95B6: 8D 00     392          HEX  8D00
95B8: 8D 8D     393 TXINST   HEX  8D8D
95BA: C9 CE D3
95BD: D4 C1 CC
95C0: CC C5 C4  394          ASC  'INSTALLED'
95C3: 8D 00     395          HEX  8D00
95C5: 87        396 TXZAP    HEX  87
95C6: DA C1 D0
95C9: A1        397          ASC  "ZAP!"
95CA: 8D 00     398          HEX  8D00
95CC: CF C6 C6  399 TXOFF    ASC  'OFF'
95CF: 8D 00     400          HEX  8D00
95D1: CB C5 D9
95D4: D7 CF D2
95D7: C4 BD BE
95DA: A0        401 TXINP    ASC  "KEYWORD=) "
95DB: 00        402          HEX  00
95DC: 87 8D 8D  403 TXERR    HEX  878D8D
95DF: C2 C1 C4
95E2: A0 CB C5
95E5: D9 D7 CF
95E8: D2 C4     404          ASC  "BAD KEYWORD"
95EA: 00        405          HEX  00
                406 *==================
                407 * Storage locations:
                408 *==================
95ED: 00 96     409 HMEMSAV  DA   $9600      ;Default
95ED: 1B FD     410 INSAV    DA   $FD1B      ;Default
95EF: BF 9B     411 RSETSAV  DA   $9BBF      ;Default
95F1: 00        412 TOKEN    HEX  00         ;Start with highlighting off
                413 TKNINDX  DS   1
                414 KEYWORD  DS   10
95FD: 00        415          HEX  00


--End assembly--

683 bytes

Errors: 0
```

全天候 CTRL 字符檢字系統

各位磁碟用家記得在APPLE 磁碟操作系統手冊中，有一個用來找出在檔案名稱中隱藏的CTRL字符的小程式嗎？忘記了，爲何特地提到這只有九行的小程式？未回答這問題前，先問你曾否試過用一個FILENAME 存起程式，後來再照着CATALOG 一字不差的照打那個F ILENAME ，却無法LOAD 它出來，電腦不斷說FILE NOT FOUND，因爲你不小心打入了一個或多個無法在螢幕上顯示出來的CTRL一字符了，但它們又實在在那裏，你要試找出它們的機會簡直是接近零，不過若利用上述提及到在磁碟系統操作手冊中刊載的小程式（見列表 1）則可輕而易舉地檢出隱藏的CT－RL 字元，所有的CTRL字元都會以閃爍的狀態下顯示出。

```
1   REM  ::::::::::::::::::
2   REM  : CTRL.CHAR.PICK :
3   REM  :      (BASIC)    :
4   REM  ::::::::::::::::::
10   DATA 201,141,240,21,201,136
20   DATA 240,17,201,128,144,13
30   DATA 201,160,176,9,72,132
40   DATA 53,56,233,64,76,249
50   DATA 253,76,240,253
60   FOR I = 768 TO 768 + 27
70   READ V: POKE I,V: NEXT I
80   POKE 54,0: POKE 55,3
90   CALL 1002
```

列表一

這個小程式是用BASIC POKE入一小段機械語言程式，令CTRL字元以閃爍方式顯示。

程式在APPLESOFT，MONITOR 甚至INTERGE 中都有效，唯載入時必須在APPL ESOFT 中，筆者覺得以閃爍狀態顯示之字體閱讀起來甚不方便，尤其在DEBUG時，令眼睛疲倦。另一項缺點，是程式用PR＃0 脫離後， 不能再以CALL 1002 或任何方便的方法再進入，加上以BASIC編寫，用起來頗不方便。但它的最大優點，是保留了CTRL－H 亦即是BACKSPACE 左箭咀不受干擾。

筆者爲此特別編寫一個強化版本的程式，程式名字叫CTRL CHAR PICK（見列表 2）。強化版本跟舊BASIC程式（列表 1）有顯著分別。首先是這系統的操作極之簡單，只要BRUN 載入系統，再打入「&」便可以，此後，任何時間均可將所有隱藏在程式裏的CTRL 字元全部用反白模態顯示出來。若需要脫離系統，可打入PR＃0 即可。要重新再進入程式，只要再次打入「&」即可。

強化版本唯一美中不足的地方，就是按左箭咀（←）時，會出現反白H，而並非令使游標向左移。（但把CTRL－H以反白顯示，編寫「無行號的REM」時很方便，各位可以LIST BEAGLE BROS 的程式來看，便會明白。）它以機械語言編寫，不影响APPLESOFT的記憶，而且，在INT－EGER 和 MONITOR 中一樣有效，但因"&"字是APP－LESOFT 專有的，故載入時必須在APPLESOFT 中，個人認爲反白的字體爲較穩定，讀起來較容易，各位不妨比較一下。

這個強化版程程式（列表 2）列表很短，是用MERLIN ASSEMBLER 寫成，懂得ASSEMBLER 的讀者可照列表 2 打入。或者直接在MONITOR 系統中鍵入左邊的機械碼（機械語言輸入方法可參考「新讀者需知」），然後用下列指令存入磁碟。

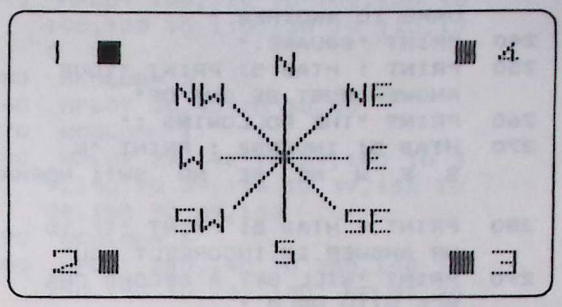
BSAVE CTRL CHAR PICK，A＄300，L＄34。

或者讀者可在同期出版的程式磁碟找到上述三個程式FILE。 ▣

```
]ASSEN
                1   ********************
                2   * CTRL.CHAR.PICK.    *
                3   * BY COMPUHOUSE 1985 *
                4   * BY P.S.WONG        *
                5   ********************
                6   *
                7   * MERLIN ASSEMBLER
                8   *
                9   *
                10  *
                11          ORG  $300
                12  *
                13  CSWL      EQU  $36              ;THE I/O HOOK
                14  SETIO     EQU  $3EA             ;DOS 3.3 I/O CONNECTOR
                15  AMPVECT   EQU  $3F5             ;& VECTOR
                16  BASIC     EQU  $D43C            ;START OF APPLESOFT
                17  CROUT     EQU  $FDFB            ;PRONTS A RETURN CHARACTER
                18  COUT2     EQU  $FDF6            ;MONITOR OUTPUT TO SCREEN
                19  *
                20  *
                21  *
                22  *   BRUN OR CALL 768 ENTRY
                23  *
0300: A9 4C     24  BRUN    LDA  #$4C               ;SET UP & VECTOR  STORE A
0302: 8D F5 03  25          STA  AMPVECT            ;'JMP SETUP' INSTRUCTION
0305: A9 10     26          LDA  #(SETUP            ;BEGINNING AT THE AMPVECTOR
0307: 8D F6 03  27          STA  AMPVECT+1          ;ADDRESS, $3F5
030A: A9 03     28          LDA  #)SETUP
030C: 8D F7 03  29          STA  AMPVECT+2
030F: 60        30          RTS             ;DONE, '&' NOW JUMPS TO SETUP
                31  *
                32  *
                33  *   SETUP THE I/O HOOKS
                34  *
0310: A9 21     35  SETUP   LDA  #(IOSTRT           ;AFTER APPLESOFT '&' JUMP HERE
0312: 85 36     36          STA  CSWL               ;SET 2 BYTE CSWL TO 2 BYTE IOSTRT
0314: A9 03     37          LDA  #)IOSTRT           ;UPPER BYTE OF IPSTAT
0316: 85 37     38          STA  CSWL+1
0318: 20 EA 03  39          JSR  SETIO              ;TELL DOS ABOUT THE CHANGE
031B: 20 FB DA  40          JSR  CROUT              ;PRINT A RETURN
031E: 4C 3C D4  41          JMP  BASIC              ;BACK TO APPLESOFT
                42  *
                43  *
                44  *   PRINT THE CHARACTER IN THE A-REG
                45  *
0321: C9 8D     46  IOSTRT  CMP  #$8D               ;RETURN ?
0323: F0 04     47          BEQ  SCRNPRT            ;IF YES BRANCH TO SCRNPRT
0325: C9 A0     48          CMP  #$A0               ;NO,  A CTRL CHAR ?
0327: 90 03     49          BCC  CONVERT            ;BRANCH TO CONVERT IF YES
0329: 4C F6 FD  50  SCRNPRT JMP  COUT2              ;NO - PRINT AS IS
032C: 69 80     51  CONVERT ADC  #$80               ;SHIFT CTRL CHAR TO INV ALPHA
```

方向概念練習
東南西北

□黃博思□



這個程式運用高解像繪圖來帮助兒童學習指南針方位，它的主要特色，是以問答遊戲的形式進行教授，寓教育於娛樂，在第一次發問時，是不會提供任何輔助貼士，但在補答時，便會提供帮助，令玩的人有興趣繼續學下去；雖然補答機會只有一個。另外，計分的顯示包括玩者的即中及補中的分數和百分比。當然，在精彩的問答遊戲中不會缺少的，是特別的音樂效果，我們的程式找到「不羈的風」做主題曲，在每答中一題時，你便會聽到這首歌的前奏。

雖然程式是爲小學同學而寫的，但坦白說，照筆者「實驗」所得，問題亦難倒不少「成年人」，最常見的錯誤，是把某甲在某乙的方向，和某乙在某甲的方向混淆了。所以，「成年人」也不妨一試。

在玩這程式前，同學宜對閱讀指南針的座標有初步認識，在本程式中用到的方位有八個，其分別爲東、南、西、北、東北、東南、西北、西南。

當ＲＵＮ本程式時，電腦會詢問你是否需要閱讀操作指引，指引內容主要是說正確的回答必須在那８個方位的範圍內，即Ｎ，Ｓ，Ｅ，Ｗ，ＮＥ，ＳＥ，ＳＷ，ＮＷ 八個。在印出指引時會播出主題曲「不羈的風」。

讀完指引，螢幕便會顯示出四塊顏色方板，分佈於螢幕四角，然後詢問你由其中的一塊方板應朝那個方向走。這裏爲了補救在單色監視器上顏色難辨之弊，方板旁都附有編號，問題亦會註明了方板的顏色及編號。

若你一答便中，就可以繼續試下一條，反之，若你答錯了，便有一次補答機會，電腦會給你一個指南針，上面寫了八個方位的應對名稱，帮助你進行補答。若你不幸補答失敗，正確的答案便會顯示，這顯示將會保留，直至你按鍵要求試下一題爲止。

若你輸入的答案不在八個方位的範圍內，則正確的答案範圍將會顯示。

若你想結束遊戲，可以不答問題，直接按RETURN，答對及補答的數量及命中率都會顯示出來，你可以選擇上訴，要求再玩一次，亦可以離開程式。

由這刻起，你不用死板的拿着指南針和地圖去學方向了，用「東南西北」吧。

本遊戲程式只有一個APPLESOFT列表，大家於打入後可用SAVE COMPASS QUIZ 存入磁碟。假如閣下購有磁碟只要RUN COMPASS QUIZ 便可。 ▣

```
 10  REM  ******************
 20  REM  *  COMPASS QUIZ  *
 30  REM  *       BY       *
 40  REM  *   COMPUHOUSE   *
 50  REM  ******************
 60  DIM P(45),T(47),D(47)
 70  HOME : VTAB 22: PRINT "** COP
     YRIGHT 1985 BY COMPUTING AGE
      **"
 80  HTAB 15: VTAB 5: PRINT "COMPA
     SS QUIZ"
 90  PRINT : PRINT : HTAB 20: PRINT
     "BY"
100  PRINT : PRINT : HTAB 13: PRINT
     "MR. DOCTOR BOWIE"
110  REM   INSTRUCTIONS ?
120  PRINT : PRINT : HTAB 5: INPUT
     "DO YOU WANT INSTRUCTIONS (Y
     /N) ?";RE$
130 RE$ =  LEFT$ (RE$,1)
140  FOR R = 0 TO 45: READ P(R): NEXT
     : FOR I = 1 TO 66: READ P: POKE
     I + 767,P: NEXT : POKE 1013,
     76: POKE 1014,0: POKE 1015,3
     : DATA 0,192,181,171,161,152
     ,144,136,128,121,114,108,102
     ,96,91,86,81,76,72,68,64,60,
     57,54,51,48,45,42,39,37,35,3
     2,31,28,27,26,24,21,21,20,19
     ,18,17
150  DATA 16,15,14,201,84,208,15,
     32,177,0,32,248,230,138,72,3
     2,183,0,201,44,240,3,76,201,
     222,32,177,0,32,248,230,104,
     134,3,134,1,133,0,170,160,1,
     132,2,173,48,192,136,208,4,1
     98,1,240,7,202,208,246,166,0
     ,208,239,165,3,133,1,198,2,2
     08,241,96
160  FOR I = 1 TO 47: READ T(I),D
     (I): NEXT
170  DATA 1,30,1,30,1,30,13,30,13
     ,30,13,60,11,60,10,30,6,30,8
     ,30,10,60,10,30,10,60,3,30,3
     ,30,6,30,6,030,6,30,5,30,6,3
     0,5,30,3,30,3,120,0,30,3,30,
     3,30,6,30,11,30,10,60,10,30,
     0,30,3,30,3,30,6,30,11,30
180  DATA 10,120,0,30,11,30,0,30,
     11,30,10,30,11,30,0,30,8,30,
     6,30,8,30,8,120
190  IF RE$ = "N" THEN 360
200  REM  INSTRUCTIONS
210  HOME : HTAB 5: VTAB 5
220  PRINT "YOU WILL BE ASKED TO
     GIVE THE "
```

方向概念練習——東南西北

```
230  PRINT "DIRECTION FROM ONE SQ
     UARE TO ANOTHER "
240  PRINT "SQUARE."
250  PRINT : HTAB 5: PRINT "YOUR
     ANSWER MUST BE ONE OF"
260  PRINT "THE FOLLOWING :"
270  HTAB 5: INVERSE : PRINT "N
     S  E  W  NE  SE  NW  SW": NORMAL

280  PRINT : HTAB 5: PRINT "IF YO
     UR ANSWER IS INCORRECT YOU"
290  PRINT "WILL GET A SECOND CHA
     NCE WITH HELP."
300  PRINT : PRINT : PRINT : PRINT
     : PRINT "PRESS <RETURN> TO E
     ND AND GET YOUR SCORE"
310  PRINT : PRINT : NORMAL : HTAB
     6: PRINT "PRESS THE SPACE BA
     R TO BEGIN": HTAB 20
320  FOR I = 1 TO 47
330 K =  PEEK ( - 16384): IF K >
     128 THEN 350
340  & TP(T(I)),D(I): NEXT : GOTO
     320
350  POKE  - 16368,0: IF K <  > 1
     60 THEN 340
360 R1 = 0:R2 = 0:W = 0:TL = 0:RI
     = 0: REM  SET COUNTER TO ZE
     RO
370  REM  DRAW BLOCKS
380  HGR : HCOLOR= 3
390  FOR X = 50 TO 60: FOR Y = 40
      TO 50
400  HPLOT X,Y
410  NEXT Y,X
420  HCOLOR= 1
430  FOR X = 50 TO 60: FOR Y = 14
     0 TO 150
440  HPLOT X,Y
450  NEXT Y,X
460  HCOLOR= 5
470  FOR X = 220 TO 230: FOR Y =
     40 TO 50
480  HPLOT X,Y
490  NEXT Y,X
500  HCOLOR= 6
510  FOR X = 220 TO 230: FOR Y =
     140 TO 150
520  HPLOT X,Y
530  NEXT Y,X
540  GOSUB 1350: REM  --DRAW NUME
     RALS
550  REM  **** MAIN PROGRAM ****
560  GOSUB 870: REM  GET QUESTION

570  GOSUB 930 REM  PRINT QUESTIO
     N
580  GOSUB 1110: REM  OBTAIN CORR
     ECT ANSWER
590  IF AN$ <  > CR$ THEN 650
600 R1 = R1 + 1
610  HOME : VTAB 23: HTAB 14: FLASH
     : PRINT "*** RIGHT ***"
620  GOSUB 1680: REM  ---BEEP MUS
     IC
630  FOR X = 1 TO 1500: NEXT : NORMAL

640  GOTO 560
650  REM  --SECOND ATTEMPT
660  HOME : VTAB 23: HTAB 13
670  & T250,100: & T250,100: REM
      ---BEEP
680  PRINT "SORRY!  TRY AGAIN": FOR
     X = 1 TO 500: NEXT
690  GOSUB 1180: REM  DRAW COMPPA
     SS
700  GOSUB 930: REM  REPEAT QUEST
     ION
710  IF AN$ = "" THEN 1440
720  GOSUB 1110: REM  OBTAIN CORR
     ECT ANSWER
730  IF AN$ <  > CR$ THEN 790
740 R2 = R2 + 1
750  HOME : VTAB 23: HTAB 19: PRINT
     "RIGHT!"
760  GOSUB 1680: REM  ---BEEP MUS
     IC
770  FOR X = 1 TO 500: NEXT
780  GOTO 370
790  REM  2 INCORRECT RESPONSES
800  & T250,100: & T250,100: REM
      ---BEEP
810 W = W + 1
820  HOME : VTAB 23: HTAB 6
830  INVERSE : PRINT "SORRY! THE
     CORRECT ANSWER IS ";CR$: NORMAL

840  HTAB 4: PRINT "PRESS THE SPA
     CE BAR TO CONTINUE": HTAB 20

850  GET X$: IF X$ <  >  CHR$ (32
     ) THEN 850
860  GOTO 370
870  REM  GENERATE A QUESTION
880 A$(1) = "WHITE(1)":A$(2) = "G
     REEN(2)":A$(3) = "BLUE(3)":A
     $(4) = "ORANGE(4)"
890 N =  INT ( RND (1) * 4 + 1)
900 P =  INT ( RND (2) * 4 + 1)
910  IF N = P THEN 900
920  RETURN
930  REM  PRINT QUESTION
940  HOME
950  VTAB 21: HTAB 8: PRINT "WHAT
      IS THE DIRECTION FROM"
960  HTAB 10: PRINT A$(N)" TO "A$
     (P)
970  HTAB 18: INPUT AN$
980  IF AN$ = "" THEN 1440
990 AN$ =  LEFT$ (AN$,2)
1000  IF  RIGHT$ (AN$,1) = " " THEN
     AN$ =  LEFT$ (AN$,1)
1010  IF AN$ <  > "N" AND AN$ <  >
     "S" AND AN$ <  > "E" AND AN$
      <  > "W" AND AN$ <  > "NE" AND
     AN$ <  > "SE" AND AN$ <  > "
     NW" AND AN$ <  > "SW" THEN 1
     030
```

方向概念練習——東南西北

```
1020  RETURN
1030  REM  --- ILLEGAL RESPONSE--
      -
1040  & T1,50: & T1,50: REM  ---B
      EEP
1050  HOME
1060  VTAB 21: PRINT "THE ANSWER
      MUST BE ONE OF THE FOLLOWING
      "
1070  HTAB 10: INVERSE : PRINT "N
      S E W NE SE NW SW": NORMAL

1080  HTAB 4: PRINT "PRESS THE SP
      ACE BAR TO CONTINUE": HTAB 2
      0
1090  GET X$: IF X$ <  >  CHR$ (3
      2) THEN 1090
1100  GOTO 930
1110  REM  ---DETERMINE CORRECT R
      ESPONSE
1120 AP$(1,2) = "S":AP$(1,3) = "S
      E":AP$(1,4) = "E"
1130 AP$(2,1) = "N":AP$(2,3) = "E
      ":AP$(2,4) = "NE"
1140 AP$(3,1) = "NW":AP$(3,2) = "
      W":AP$(3,4) = "N"
1150 AP$(4,1) = "W":AP$(4,2) = "S
      W":AP$(4,3) = "S"
1160 CR$ = AP$(N,P)
1170  RETURN
1180  HCOLOR= 6
1190  HPLOT 140,65 TO 140,125
1200  HPLOT 110,95 TO 170,95
1210  HPLOT 115,70 TO 165,120
1220  HPLOT 165,70 TO 115,120
1230  HPLOT 136,55 TO 136,45 TO 1
      44,55 TO 144,45
1240  HPLOT 144,135 TO 136,135 TO
      136,140 TO 144,140 TO 144,14
      5 TO 136,145
1250  HPLOT 188,90 TO 180,90 TO 1
      80,100 TO 188,100: HPLOT 180
      ,95 TO 184,95
1260  HPLOT 90,90 TO 90,100 TO 95
      ,95 TO 100,100 TO 100,90
1270  HPLOT 170,70 TO 170,60 TO 1
      78,70 TO 178,60
1280  HPLOT 190,60 TO 182,60 TO 1
      82,70 TO 190,70: HPLOT 182,6
      5 TO 186,65
1290  HPLOT 178,120 TO 170,120 TO
      170,125 TO 178,125 TO 178,13
      0 TO 170,130
1300  HPLOT 190,120 TO 182,120 TO
      182,130 TO 190,130: HPLOT 18
      2,125 TO 186,125
1310  HPLOT 88,70 TO 88,60 TO 96,
      70 TO 96,60
1320  HPLOT 100,60 TO 100,70 TO 1
      05,65 TO 110,70 TO 110,60
1330  HPLOT 96,120 TO 88,120 TO 8
      8,125 TO 96,125 TO 96,130 TO
      88,130
1340  HPLOT 100,120 TO 100,130 TO
      105,125 TO 110,130 TO 110,12
      0
1350  HCOLOR= 7
1360  HPLOT 30,40 TO 30,50
1370  HCOLOR= 1
1380  HPLOT 29,142 TO 29,140 TO 3
      9,140 TO 39,146 TO 39,146 TO
      29,150 TO 39,150
1390  HCOLOR= 6
1400  HPLOT 240,140 TO 250,140 TO
      250,150 TO 240,150: HPLOT 24
      4,145 TO 250,145
1410  HCOLOR= 5
1420  HPLOT 250,48 TO 240,48 TO 2
      48,40 TO 246,50
1430  RETURN
1440  REM  SOCRE SUMMARY
1450 TL = R1 + R2 + W:RI = R1 + R
      2
1460  TEXT : HOME : VTAB 3: HTAB
      15
1470  PRINT "YOUR SCORE"
1480  HTAB 15: PRINT "==========="
      : REM  10 = 'S
1490  PRINT : PRINT : PRINT
1500  IF TL = 0 THEN  HTAB (8): PRINT
      "NO QUESTIONS WERE ATTEMPTED
      .": GOTO 1590
1510  PRINT "ON THE FIRST TRY"
1520  PRINT "----------------": REM
       16 - 'S
1530  PRINT
1540  HTAB 5: PRINT R1" OUT OF "T
      L" FOR " INT (R1 * 100 / TL)
      " %"
1550  PRINT : PRINT
1560  PRINT "ON THE FIRST AND SEC
      OND TRY"
1570  PRINT "--------------------
      --------": REM  27 -'S
1580  HTAB 5: PRINT RI" OUT OF "T
      L" FOR " INT (RI * 100 / TL)
      " %"
1590  PRINT : PRINT : PRINT
1600  HTAB 5: PRINT "PRESS THE SP
      ACE BAR TO CONTINUE"
1610  PRINT : HTAB 8: PRINT "PRES
      S ANY KEY TO STOP."
1620  FOR I = 1 TO 47
1630 K =  PEEK ( - 16384): IF K >
      128 THEN 1650
1640  & TP(T(I)),D(I): NEXT : GOTO
      1620
1650  POKE  - 16368,0: IF K <  >
      160 THEN 1640
1660  HOME : END
1670  REM  ---SOUND 1
1680  & TP(18),60: & TP(18),30: &
      TP(18),30: & TP(18),60: & TP
      (18),30: & TP(17),60: & TP(1
      8),255: RETURN
```

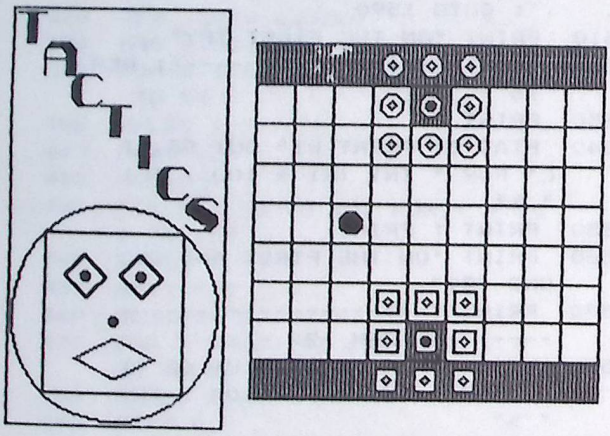
戰略跳棋 TACTICS

■陳文緯■

下棋最講究戰術。這是一副訓練戰術的棋，定可活動一下大家的腦筋，只供兩人玩。

開始時，電腦會清除螢幕，畫出TACTICS 字樣及其標誌，並奏出一段意大利民歌。跟着畫９×９的棋盤，而其綠色的格子爲陣地。玩者一的棋子是以圓形代表，位於盤的上部份；玩者二的棋子是以正方形代表，位於盤的下部份。每玩者共有９隻棋子，其中８隻的中心都是菱形的，稱爲Ｒ棋；剩下的一隻棋子中心是圓形的，稱爲Ｃ棋。玩者若能最先將自己的Ｃ棋行到對方十格陣地中其中一格便算是勝利了。

玩者一先行，首先以ＩＪＫＭ行至要移動的棋子上，跟着按ＲＥＴＵＲＮ鍵（稍後如果發覺選錯了可按ＥＳＣ鍵來重新選擇）。各玩者每次只可移動一隻棋子，而移動後每格只可存有一隻棋子。



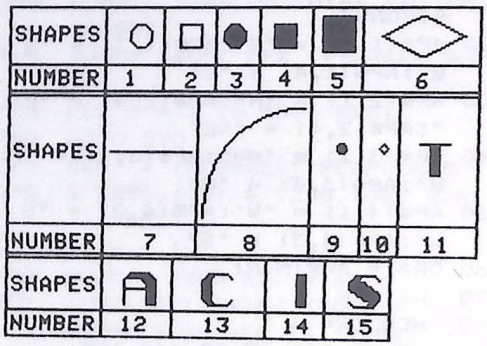
Ｃ棋的移動方法是跟跳棋大同小異的，只要有一隻Ｒ棋在它的旁邊其中一格（八邊），它便可跳過那隻Ｒ棋而到另一面。Ｃ棋可以連跳，所以在用ＩＪＫＭＵＯＮ，鍵控制完它後要按ＲＥＴＵＲＮ鍵停止跳動。因Ｃ棋不能吃別的棋子，所以不可跳至已有棋子的格子。

Ｒ棋可向前（即向着對方陣地）、左或右移動，吃棋時需向左前方（斜行）或右前方（斜行）移動。玩者可用ＩＪＫＭＵＯＮ，八鍵控制棋子。如果吃了Ｒ棋，便可減少對方一隻Ｒ棋。若吃下了Ｃ棋，便可把這隻Ｃ棋任意放在棋盤的一格，方法是用ＩＪＫＭ四鍵控制在棋盤旁出現的箭咀，然後按ＲＥＴＵＲＮ鍵把對方的Ｃ棋放在哪裏。

若把自己的Ｒ棋移至敵方陣地，便能返回自己的陣地，方法是用ＩＪＫＭ鍵控制在棋盤旁出現的箭咀，然後按Ｒ—ＥＴＵＲＮ把這Ｒ棋放下。

本程式共有兩個檔案：①ＢＡＳＩＣ主程式ＴＡＣＴ—ＩＣＳ②機械碼主程式TACTICS SHAPES（A$6000，L$30D）輸入方法可參考第十七期的「機械碼輸入方法」，是以ＰＩＸＩＴ繪成的圖形表，可參看圖一。主程式ＴＡＣＴＩＣＳ的程式分析在下：

FIG.1 圖一

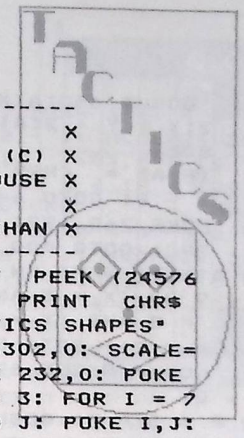


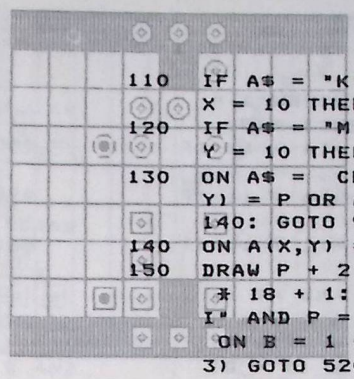
10　載入圖形表
20　INITIAZATION
30　劃出螢幕右邊和播放音樂
60　劃出棋盤和棋子
90　選擇棋子之檢查按制程序
150　菱形棋子之檢查按制程序
260　箭咀指標之檢查按制程序
390　圓形棋子之檢查按制程序
510　檢查菱形棋子是否到達別方陣地
520　檢查圓形棋被吃
530　ＬＯ—ＲＥＳ副程式
560　程式後部
590　清除某格
600　劃了玩者一的菱形棋子
610　劃了玩者二的菱形棋子
620　劃了玩者一的圓形棋子
630　劃了玩者二的圓形棋子
640　劃了棋子
650　清除某格的棋子
660　發聲副程式
670　音樂資料，包括音高和音長

主程式用了十一期的ＬＩＮＥ—ＣＲＵＮＣＨＥＲ和ＡＰＰ—ＬＥＳＯＦＴ ＴＯＯＬ ＫＩＴ的ＲＥＮＵＭＢＥＲ，使程式既可節省空間，又可使ＬＩＳＴＩＮＧ 更美觀和加快運行速度。

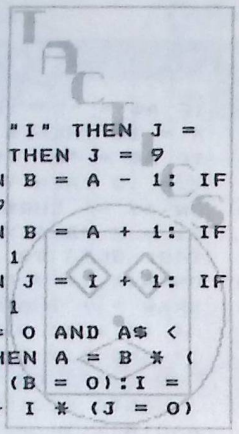
■

戰略跳棋 TACTICS

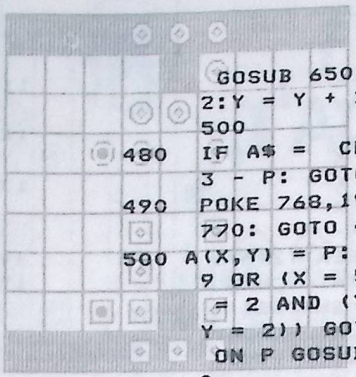




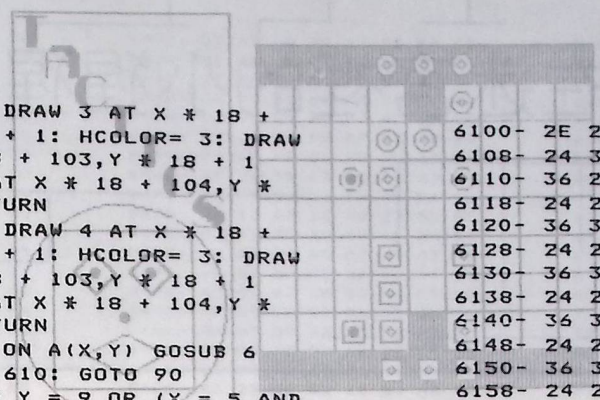
```
1   REM  ----------------
2   REM  X TACTICS       X
3   REM  X COPYRIGHT (C) X
4   REM  X BY COMPUHOUSE X
5   REM  X BY            X
6   REM  X MANVILLE CHAN X
7   REM  ----------------
10   DIM A(10,10): IF  PEEK (24576
     ) <  > 16 THEN  PRINT  CHR$
     (4);"BLOAD TACTICS SHAPES"
20  HGR : POKE  - 16302,0: SCALE=
     1: ROT= 0: POKE 232,0: POKE
     233,96: HCOLOR= 3: FOR I = 7
     70 TO 790: READ J: POKE I,J:
      NEXT : FOR I = 1 TO 7: READ
     J: DRAW J AT I * 14 - 12,I *
     14 - 12: NEXT
30  ROT= 16: XDRAW 9 AT 101,138: ROT=
     32: XDRAW 9 AT 53,189: ROT=
     48: XDRAW 9 AT 2,141: ROT= 0
     : XDRAW 9 AT 50,90: HPLOT 18
     ,102 TO 85,102 TO 85,176 TO
     18,176 TO 18,102
40  DRAW 10 AT 48,139: HPLOT 38,1
     12 TO 28,122 TO 38,132 TO 48
     ,122 TO 38,112 TO 38,112: HPLOT
     65,112 TO 55,122 TO 65,132 TO
     75,122 TO 65,112: DRAW 10 AT
     34,118: DRAW 10 AT 61,118: HPLOT
     0,0 TO 0,191 TO 104,191 TO 1
     04,0 TO 0,0
50  FOR I = 1 TO 11: FOR J = 7 TO
     8: XDRAW J AT 33,160: READ A
     ,B: POKE 768,A: POKE 769,B *
     64: CALL 770: XDRAW J AT 33,
     160: NEXT J,I: DRAW 7 AT 33,
     160
60  FOR I = 1 TO 10: HPLOT I * 18
      + 99,15 TO I * 18 + 99,177:
      HPLOT 117,I * 18 - 3 TO 279
     ,I * 18 - 3: NEXT : FOR I =
     1 TO 9:X = I:Y = 1: GOSUB 59
     0:Y = 9: GOSUB 590: NEXT :X =
     5:Y = 2: GOSUB 590:Y = 8: GOSUB
     590
70  FOR I = 4 TO 6:A(I,1) = 3:X =
     I:Y = 1: GOSUB 600:A(I,3) =
     3:Y = 3: GOSUB 600:A(I,7) =
     4:Y = 7: GOSUB 610:A(I,9) =
     4:Y = 9: GOSUB 610: NEXT
80 A(4,2) = 3:X = 4:Y = 2: GOSUB
     600:A(6,2) = 3:X = 6: GOSUB
     600:A(6,8) = 4:Y = 8: GOSUB
     610:A(4,8) = 4:X = 4: GOSUB
     610:A(5,2) = 1:X = 5:Y = 2: GOSUB
     620:A(5,8) = 2:Y = 8: GOSUB
     630:X = 1:Y = 1:P = 1:A = 0:
     B = 0
90  XDRAW P + 2 AT X * 18 + 103,Y
     * 18 + 1: XDRAW P + 2 AT X *
     18 + 103,Y * 18 + 1: ON  PEEK
     ( - 16384) < 128 GOTO 90: GET
     A$: IF A$ = "I" THEN Y = Y -
     1: IF Y = 0 THEN Y = 9
100  IF A$ = "J" THEN X = X - 1: IF
     X = 0 THEN X = 9
110  IF A$ = "K" THEN X = X + 1: IF
     X = 10 THEN X = 1
120  IF A$ = "M" THEN Y = Y + 1: IF
     Y = 10 THEN Y = 1
130  ON A$ =  CHR$ (13) AND (A(X,
     Y) = P OR A(X,Y) = P + 2) GOTO
     140: GOTO 90
140  ON A(X,Y) = P GOTO 390
150  DRAW P + 2 AT X * 18 + 103,Y
      * 18 + 1: GET A$: IF A$ = "
     I" AND P = 2 THEN B = Y - 1:
      ON B = 1 OR (X = 5 AND B =
     3) GOTO 520: ON A(X,B) > 0 GOTO
     660:A(X,Y) = 0: GOSUB 650:Y =
     B
160  IF A$ = "J" THEN A = X - 1: ON
     A = 0 OR A(A,Y) > 0 GOTO 660
     :A(X,Y) = 0: GOSUB 650:X = A

170  IF A$ = "K" THEN A = X + 1: ON
     X = 10 OR A(A,Y) > 0 GOTO 66
     0:A(X,Y) = 0: GOSUB 650:X =
     A
180  IF A$ = "M" AND P = 1 THEN B
      = Y + 1: ON B = 9 OR (X = 5
      AND B = 8) GOTO 520: ON A(X
     ,B) > 0 GOTO 660:A(X,Y) = 0:
      GOSUB 650:Y = B
190  IF A$ = "U" AND (A(X - 1,Y -
     1) = 3 - P + 2 OR A(X - 1,Y -
     1) = 3 - P) AND P = 2 THEN A
      = X - 1:B = Y - 1: GOTO 250

200  IF A$ = "O" AND (A(X + 1,Y -
     1) = 3 - P + 2 OR A(X + 1,Y -
     1) = 3 - P) AND P = 2 THEN A
      = X + 1:B = Y - 1: GOTO 250

210  IF A$ = "N" AND (A(X - 1,Y +
     1) = 3 - P + 2 OR A(X - 1,Y +
     1) = 3 - P) AND P = 1 THEN A
      = X - 1:B = Y + 1: GOTO 250

220  IF A$ = "," AND (A(X + 1,Y +
     1) = 3 - P + 2 OR A(X + 1,Y +
     1) = 3 - P) AND P = 1 THEN A
      = X + 1:B = Y + 1: GOTO 250

230  ON A$ =  CHR$ (27) GOTO 640:
      ON A = 0 AND B = 0 GOTO 660

240  ON P GOSUB 600,610:A(X,Y) =
     P + 2:A = 0:B = 0:P = 3 - P:
      GOTO 90
250  ON A(A,B) = 3 - P GOTO 510: GOSUB
     650: FOR I = 1 TO 10: POKE 7
     68,I * 2: POKE 769,I: CALL 7
     70: NEXT :A(X,Y) = 0:X = A:Y
      = B: GOSUB 650: GOTO 240
260  GOSUB 650:A(X,Y) = 0:X = A:Y
      = B: GOSUB 650: ON P GOSUB
     610,600:A(X,Y) = 3 - P + 2
270 A = 1:I = 1:B = 0:J = 0
280  XDRAW 5 AT A * 18 + 106,178:
      ROT= 16: XDRAW 5 AT 116,I *
     18 + 3: ROT= 0
```



```
290  GET A$: IF A$ = "I" THEN J =
     I - 1: IF J = 0 THEN J = 9
300  IF A$ = "J" THEN B = A - 1: IF
     B = 0 THEN B = 9
310  IF A$ = "K" THEN B = A + 1: IF
     B = 10 THEN B = 1
320  IF A$ = "M" THEN J = I + 1: IF
     J = 10 THEN J = 1
330  IF B = 0 AND J = 0 AND A$ <
      > CHR$ (13) THEN A = B * (
     B <  > 0) + A * (B = 0):I =
     J * (I <  > 0) + I * (J = 0)
     : GOTO 290
340  XDRAW 5 AT A * 18 + 106,178:
      ROT= 16: XDRAW 5 AT 116,I *
     18 + 3: ROT= 0: IF A$ =  CHR$
     (13) THEN B = 1:J = 1
350  IF B = 0 THEN I = J
360  IF J = 0 THEN A = B
370 B = 0:J = 0: ON A$ <  >  CHR$
     (13) GOTO 280: IF A(A,I) > 0
      THEN  POKE 768,192: POKE 76
     9,128: CALL 770: GOTO 270
380 X = I:Y = A: RETURN
390  DRAW P + 2 AT X * 18 + 103,Y
      * 18 + 1
400  GET A$: IF A$ = "I" AND Y >
     2 THEN  IF A(X,Y - 2) = 0 AND
     A(X,Y - 1) > 2 THEN  GOSUB 6
     50:A(X,Y) = 0:Y = Y - 2: GOTO
     500
410  IF A$ = "J" AND X > 2 THEN  IF
     A(X - 2,Y) = 0 AND A(X - 1,Y
     ) > 2 THEN  GOSUB 650:A(X,Y)
      = 0:X = X - 2: GOTO 500
420  IF A$ = "K" AND X < 8 THEN  IF
     A(X + 2,Y) = 0 AND A(X + 1,Y
     ) > 2 THEN  GOSUB 650:A(X,Y)
      = 0:X = X + 2: GOTO 500
430  IF A$ = "M" AND Y < 8 THEN  IF
     A(X,Y + 2) = 0 AND A(X,Y + 1
     ) > 2 THEN  GOSUB 650:A(X,Y)
      = 0:Y = Y + 2: GOTO 500
440  IF A$ = "U" AND X > 2 AND Y >
     2 THEN  IF A(X - 2,Y - 2) =
     0 AND A(X - 1,Y - 1) > 2 THEN
      GOSUB 650:A(X,Y) = 0:X = X -
     2:Y = Y - 2:A(X,Y) = P: GOTO
     500
450  IF A$ = "O" AND X < 8 AND Y >
     2 THEN  IF A(X + 2,Y - 2) =
     0 AND A(X + 1,Y - 1) > 2 THEN
      GOSUB 650:A(X,Y) = 0:X = X +
     2:Y = Y - 2:A(X,Y) = P: GOTO
     500
460  IF A$ = "N" AND X > 2 AND Y <
     8 THEN  IF A(X - 2,Y + 2) =
     0 AND A(X - 1,Y + 1) > 2 THEN
      GOSUB 650:A(X,Y) = 0:X = X -
     2:Y = Y + 2:A(X,Y) = P: GOTO
     500
470  IF A$ = "," AND X < 8 AND Y <
     8 THEN  IF A(X + 2,Y + 2) =
     0 AND A(X + 1,Y + 1) > 2 THEN
```



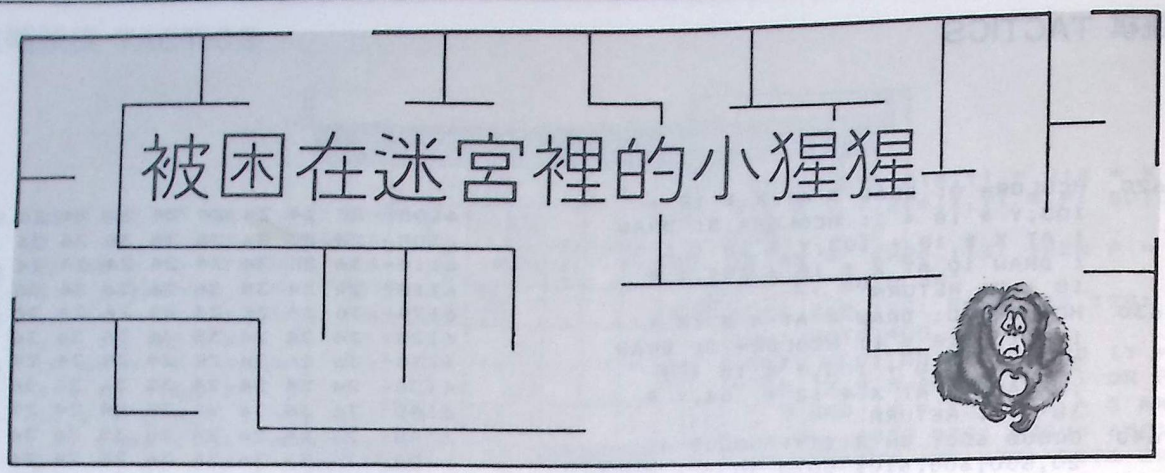
```
      GOSUB 650:A(X,Y) = 0:X = X +
     2:Y = Y + 2:A(X,Y) = P: GOTO
     500
480  IF A$ =  CHR$ (13) THEN P =
     3 - P: GOTO 640
490  POKE 768,192: POKE 769,128: CALL
     770: GOTO 400
500 A(X,Y) = P: ON P = 1 AND (Y =
     9 OR (X = 5 AND Y = 8)) OR P
      = 2 AND (Y = 1 OR (X = 5 AND
     Y = 2)) GOTO 530: GOSUB 650:
      ON P GOSUB 620,630: GOTO 39
     0
510 P = 3 - P: GOSUB 260:A(A,I) =
     P:X = A:Y = I: GOSUB 650: ON
     A(X,Y) GOSUB 620,630,600,610
     : ON (Y = 1 AND P = 2) OR (Y
      = 9 AND P = 1) OR (X = 5 AND
     Y = 2 AND P = 2) OR (X = 5 AND
     Y = 8 AND P = 1) GOTO 530: GOTO
     90
520  GOSUB 650:A(X,Y) = 0:X = A:Y
      = B: GOSUB 270:A(A,I) = P +
     2:X = A:Y = I: ON (P = 1 AND
     (I = 1 OR (I = 2 AND A = 5))
     ) OR (P = 2 AND (I = 9 OR (I
      = 8 AND A = 5))) GOTO 240: POKE
     768,192: POKE 769,128: CALL
     770: GOTO 520
530  POKE  - 16368,0: CALL  - 199
     8: COLOR= 15: FOR I = 6 TO 2
     4: HLIN 15,25 AT I: NEXT : ON
     P = 2 GOTO 550: COLOR= 0: VLIN
     6,11 AT 15: VLIN 6,11 AT 25:
      VLIN 19,24 AT 15: VLIN 19,2
     4 AT 25: VLIN 6,9 AT 16: VLIN
     6,9 AT 24
540  VLIN 6,7 AT 17: VLIN 6,7 AT
     23: VLIN 21,24 AT 16: VLIN 2
     1,24 AT 24: VLIN 23,24 AT 17
     : VLIN 23,24 AT 23
550  POKE  - 16298,0: ON  PEEK ( -
     16384) > 127 GOTO 560: POKE
      - 16297,0: GOTO 550
560  TEXT : HOME : VTAB 10: PRINT
     "DO YOU WANT TO PLAY AGAIN ?
      ( Y OR N ) ";
570  GET A$: IF A$ = "Y" THEN  RUN

580  ON A$ <  > "N" GOTO 570: HOME
     : END
590  HCOLOR= 0: DRAW 6 AT X * 18 +
     100,Y * 18 - 2: HCOLOR= 1: DRAW
     6 AT X * 18 + 100,Y * 18 - 2
     : HCOLOR= 3: RETURN
600  HCOLOR= 0: DRAW 3 AT X * 18 +
     103,Y * 18 + 1: HCOLOR= 3: DRAW
     1 AT X * 18 + 103,Y * 18 + 1
     : DRAW 11 AT X * 18 + 103,Y *
     18 + 1: RETURN
610  HCOLOR= 0: DRAW 4 AT X * 18 +
     103,Y * 18 + 1: HCOLOR= 3: DRAW
     2 AT X * 18 + 103,Y * 18 + 1
     : DRAW 11 AT X * 18 + 103,Y *
     18 + 1: RETURN
```



```
620   HCOLOR= 0: DRAW 3 AT X * 18 +
      103,Y * 18 + 1: HCOLOR= 3: DRAW
      1 AT X * 18 + 103,Y * 18 + 1
      : DRAW 10 AT X * 18 + 104,Y *
      18 + 2: RETURN
630   HCOLOR= 0: DRAW 4 AT X * 18 +
      103,Y * 18 + 1: HCOLOR= 3: DRAW
      2 AT X * 18 + 103,Y * 18 + 1
      : DRAW 10 AT X * 18 + 104,Y *
      18 + 2: RETURN
640   GOSUB 650: ON A(X,Y) GOSUB 6
      20,630,600,610: GOTO 90
650   ON Y = 1 OR Y = 9 OR (X = 5 AND
      Y = 2) OR (X = 5 AND Y = 8) GOTO
      590: HCOLOR= 0: DRAW 6 AT X *
      18 + 100,Y * 18 - 2: HCOLOR=
      3: RETURN
660   POKE 768,192: POKE 769,180: CALL
      770: POKE  - 16368,0: GOTO 1
      50
670   DATA 173,48,192,136,208,5,20
      6,1,3,240,9,202,208,245,174,
      0,3,76,2,3,96,12,13,14,12,15
      ,14,16,64,1,72,1,76,1,86,2,9
      6,0,96,1,86,1,76,1,96,1,102,
      2,115,0,72,1,76,1,86,1,64,1,
      76,1,86,1,96,1,86,2,76,1,86,
      1,96,0
```

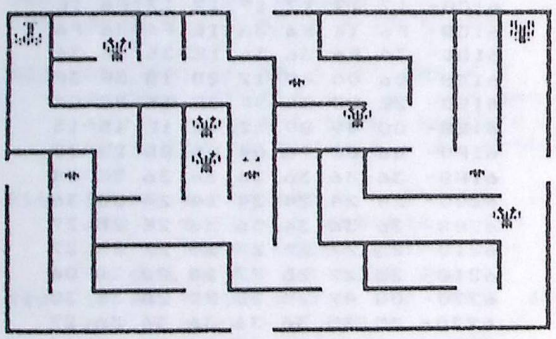
```
 ********************
 * TACTICS SHAPES *
 *  A$6000,L$30D  *
 ********************
6000- 10 00 22 00 38 00 4D 00
6008- 83 00 C1 00 D6 00 60 01
6010- 98 01 AD 01 DA 01 E9 01
6018- F5 01 21 02 64 02 97 02
6020- C0 02 49 2D 2D 15 15 15
6028- 36 36 1E 1E 1E 3F 3F 1C
6030- 1C 1C 24 24 0C 0C 04 00
6038- 2D 2D 2D 2D 2D 36 36 36
6040- 36 36 3F 3F 3F 3F 3F 24
6048- 24 24 24 24 00 49 2D 2D
6050- 15 3F 3F 3F 17 2D 2D 2D
6058- 2D 15 3F 3F 3F 3F 3F 2E
6060- 2D 2D 2D 2D 35 3F 3F 3F
6068- 3F 3F 2E 2D 2D 2D 2D 35
6070- 3F 3F 3F 3F 3F 15 2D 2D
6078- 2D 2D 17 3F 3F 3F 15 2D
6080- 2D 04 00 2D 2D 2D 2D 2D
6088- 3E 3F 3F 3F 3F 37 2D 2D
6090- 2D 2D 2D 3E 3F 3F 3F 3F
6098- 37 2D 2D 2D 2D 2D 3E 3F
60A0- 3F 3F 3F 37 2D 2D 2D 2D
60A8- 2D 3E 3F 3F 3F 3F 37 2D
60B0- 2D 2D 2D 2D 3E 3F 3F 3F
60B8- 3F 37 2D 2D 2D 2D 2D 04
60C0- 00 49 3E 0D 2E 1F 3F 3E
60C8- 0D 2D 2D 9F 36 36 3E 24
60D0- 24 3C 36 36 26 00 31 36
60D8- 36 36 36 36 36 36 2E 24
60E0- 24 24 24 24 24 24 24 35
60E8- 36 36 36 36 36 36 36 2E
60F0- 24 24 24 24 24 24 24 24
60F8- 35 36 36 36 36 36 36 36
6100- 2E 24 24 24 24 24 24 24
6108- 24 35 36 36 36 36 36 36
6110- 36 2E 24 24 24 24 24 24
6118- 24 24 35 36 36 36 36 36
6120- 36 36 2E 24 24 24 24 24
6128- 24 24 24 35 36 36 36 36
6130- 36 36 36 2E 24 24 24 24
6138- 24 24 24 24 35 36 36 36
6140- 36 36 36 36 2E 24 24 24
6148- 24 24 24 24 24 35 36 36
6150- 36 36 36 36 36 2E 24 24
6158- 24 24 24 24 24 24 07 00
6160- 2D 28 45 2B 45 2B 45 2B
6168- 45 2B 45 2B 45 2B 45 2B
6170- 45 2B 2D 15 AD AD AD AD
6178- AD AD AD AD AD BD 3B 17 BF
6180- BF BF BF BF BF BF 3F E7
6188- 3F 38 47 3B 47 3B 47 3B
6190- 47 3B 47 3B 47 3B 04 00
6198- 2D 2D 2D 2D 2D 2D 2D 2D
61A0- 2D 2D 2D 2D 2D 2D 2D 2D
61A8- 2D 2D 2D 2D 00 3F 3F 3F
61B0- 3F 17 3F BF 3F 17 3F 17
61B8- BF BF BF 17 BF 17 17 BF
61C0- 17 17 17 17 17 17 F6 1E
61C8- F6 1E F6 36 1E F6 36 F6
61D0- 36 F6 36 36 1E 36 36 36
61D8- 06 00 49 12 2D 15 3F 3F
61E0- 2E 2D 35 3F 3F 15 2D 04
61E8- 00 49 89 12 1E 1E 15 15
61F0- 0C 0C 1C 04 00 2D 2D 35
61F8- 36 36 36 36 36 36 2E 24
6200- 24 24 24 24 24 24 35 36
6208- 36 36 36 36 36 2E 25 27
6210- 25 27 25 27 25 27 25 27
6218- 25 27 25 27 2D 2D 2D 06
6220- 00 49 2D 2D 2D 2D 35 35
6228- 35 35 36 36 36 36 36 27
6230- 24 24 24 24 E4 36 36 36
6238- 36 36 3E 27 25 27 25 27
6240- 25 27 25 27 25 3F 3C 0D
6248- 25 27 37 27 37 27 37 27
6250- 37 27 37 BF 36 36 2D 2D
6258- 2D 2D 2D DF DB DB 13 36
6260- 36 36 04 00 49 2D 2D 2D
6268- 2D AD 15 DC DB DB 3B 3F
6270- 3E 0D 35 3F 3F 2E 2D 35
6278- 3F 3F 2E 2D 35 3F 3F 2E
6280- 2D 35 3F 3F 2E 2D 35 3F
6288- 3F 15 2D 35 3F 4D 3E 6F
6290- 29 2D 2D 2D 28 20 00 49
6298- 09 36 36 36 36 36 36 36
62A0- 25 24 24 24 24 24 24 2C
62A8- 36 36 36 36 36 36 36 25
62B0- 24 24 24 24 24 24 2C 36
62B8- 36 36 36 36 36 36 04 00
62C0- 92 36 B6 92 AA 15 2D 2D
62C8- 2D 2D 25 3F 4D 3C 27 3F
62D0- 27 27 3F 3C 3C 27 24 24
62D8- 0C 36 36 2E 24 24 24 35
62E0- 36 36 36 35 2D 2E 2E 2D
62E8- 2E 35 0C 3C 27 2D 3C 3F
62F0- 3E 27 25 2D FD 38 3C 27
62F8- 27 3F 2E 3E 2E 2E 35 0C
6300- 3C 27 1C 18 40 23 2D 2D
6308- 2D AD 15 04 00 FF
```

被困在迷宮裡的小猩猩



BONGO DASH是一個快速的動感遊戲，有不同的難度可供選擇。主角BONGO是一隻可憐的小猩猩，牠被困在一系列由機械人守衛着的迷宮裏，有賴你帶牠逃出生天。

執行此遊戲，打入RUN BONGO DASH便可。這時可以看到一個小小的介紹，再按一下＜RETURN＞鍵就可正式進行遊戲了。至於控制小猩猩的移動，是用A，Z及左右箭咀鍵分別代表上下左右，不過如有需要，可以自行另定控制鍵。另外，以＜ESC＞來暫停遊戲及＜CTRL－S＞選擇有沒有聲音。



這個遊戲的目的是一方面逃避機械人的追逐，而另一方面收集在畫面上的7個椰子（白色的那些），做了以上的工作後走向左右面的出口便可升LEVEL。注意上下面的出口只供機械人作隧道之用，BONGO是不能穿過的。畫面上還不時出現一些物體如電腦、鑽石、蘋果等，這些都會爲你帶來額外的獎分。

要對付機械人，可使其撞向你放置的樹裏，而令它自行消滅。遊戲開始的時候，你擁有5株樹，而每個迷宮都有一株樹以作補充（在右下角裏）。按＜SPACE＞鍵來放置一株樹，而＜RETURN＞鍵則是把它除去。這個遊戲的說明到此爲止，至於取勝與否，則要看各位的技巧了。

本文有三個列表，第一個名爲BONGO DASH 的是BASIC主程式，只要小心鍵入再SAVE起便可。其餘兩個是機械碼列表，小心鍵入後可分別用以下指令存入碟：

列表二 BSAVE BONGO.SHAPES，A$4000,L$12F

列表三 BSAVE BONGO.OBJ，A$8000,L$97E

鍵入機械語言列表的方法可參考第十七期的「機械碼輸入方法」一文。

```
*****************
* BONGO.SHAPE   *
* A$4000,L$12F  *
*****************
4000- 02 0A E0 81 A8 85 E0 81
4008- AA 95 AA 95 A2 91 A8 85
4010- AA 95 8A 94 8A 94 02 0A
4018- 50 02 70 03 50 02 60 01
4020- 54 0A 55 2A 51 22 49 24
4028- 1C 0E 08 04 02 0A 00 00
4030- 00 00 A0 81 E8 85 B8 87
4038- A0 81 00 00 00 00 00 00
4040- 00 00 02 0A 88 44 22 90
4048- A8 05 00 D1 AA 55 00 81
4050- A2 05 28 D1 82 01 20 91
4058- 02 0A 00 00 20 90 A8 05
4060- 00 95 A8 15 20 85 A0 05
4068- 28 91 00 01 00 00 02 0A
4070- 7C 0F 22 11 2A 15 2A 15
4078- 22 11 7C 0F 1E 1E 2B 35
4080- 55 2A 7F 3F 00 00 00 02
4088- 0A 00 00 00 01 48 90 00
4090- 21 12 85 A4 10 00 A2 84
4098- 11 10 82 00 00 02 0A 08
40A0- 11 22 45 28 14 02 45 A2
40A8- C1 E0 81 A0 81 E0 81 A0
40B0- 81 E8 85 02 0A 30 00 60
40B8- 00 14 0A AA 95 AA 85 D4
40C0- 82 D4 8A 28 05 28 05 10
40C8- 02 02 0A 00 00 00 00 00
40D0- 00 10 02 54 08 54 08 54
40D8- 0A 54 0A 50 02 00 00 02
40E0- 0A 00 00 00 00 B8 97 EE
40E8- DD BA F7 EE DD BA F7 EE
40F0- DD B8 97 00 00 00 00 00
40F8- 00 00 00 00 00 00 00 00
4100- 02 0A 0A 50 20 05 C0 82
4108- C8 92 D4 AA EA D6 D2 CA
4110- C0 82 A8 95 0A 50 02 0A
4118- 00 00 00 00 D8 8A AC 95
4120- D6 AA AC 95 D8 8A B0 85
4128- E0 82 00 81 00 00 00 FF
```

```
*****************
*   BONGO.OBJ   *
* A$8000,L$97E  *
*****************
8000- 4C 06 80 4C B3 85 F8 A5
8008- FC 18 69 01 85 FC D8 20
8010- C8 85 A9 00 85 09 85 FE
8018- 8D 21 50 8D 22 50 8D 06
8020- 03 8D 09 03 8D 0E 03 8D
8028- 0F 03 8D 14 03 A0 07 99
8030- 14 70 88 D0 FA A5 F9 8D
8038- 25 50 A5 FA 8D 26 50 A2
8040- 00 20 0A 85 A9 07 8D 27
8048- 50 AC 27 50 B9 00 50 F0
8050- 11 B9 00 70 8D 25 50 B9
8058- 0A 70 8D 26 50 A2 01 20
8060- 0A 85 AC 27 50 B9 1E 70
8068- 8D 25 50 B9 28 70 8D 26
8070- 50 A2 02 20 0A 85 CE 27
8078- 50 D0 CE A9 14 8D 07 03
8080- 8D 10 03 A9 03 8D 0D 03
8088- 8D 06 03 20 36 82 20 FB
8090- 82 20 9B 88 A9 0F 85 07
8098- 85 06 A9 14 8D 27 50 20
80A0- 4D 86 C6 06 D0 F9 A9 0F
80A8- 85 06 C6 07 CE 27 50 D0
80B0- EE 20 76 81 20 2F 89 20
80B8- 36 82 20 1F 84 20 68 86
80C0- 20 8E 84 20 FB 82 20 F3
80C8- 86 20 5C 87 20 E5 87 20
80D0- 3F 88 20 9B 88 20 E0 88
80D8- A5 06 20 A8 FC A5 09 F0
80E0- D0 A9 F0 8D 11 03 20 F3
80E8- 86 CE 11 03 D0 F8 60 AC
80F0- 2A 50 B9 00 60 85 01 B9
80F8- 00 61 18 6D 2B 50 85 00
8100- A0 00 60 BD 5A 81 85 02
8108- BD 68 81 85 03 A0 00 B1
8110- 02 8D 2E 50 8D 29 50 C8
8118- B1 02 8D 2C 50 E6 02 E6
8120- 02 AD 26 50 8D 2A 50 AD
8128- 25 50 8D 2B 50 A2 00 60
8130- 20 03 81 20 EF 80 B1 00
8138- 25 7F 41 02 C1 02 D0 18
8140- E6 02 C8 CE 2E 50 D0 EE
8148- AD 29 50 8D 2E 50 EE 2A
8150- 50 CE 2C 50 D0 DD 18 60
8158- 38 60 00 16 2C 42 58 6E
8160- 87 9D B3 C9 DF 00 16 6E
8168- 40 40 40 40 40 40 40 40
8170- 40 40 40 41 41 40 AD 00
8178- C0 CD 00 03 D0 06 A9 F9
```

被困在迷宫裡的小猩猩

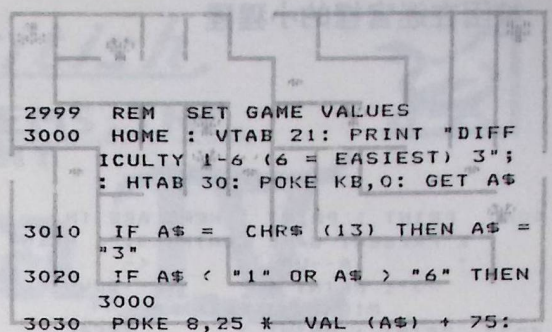
```
8180- 8D 22 50 60 CD 01 03 D0
8188- 06 A9 07 8D 22 50 60 CD
8190- 02 03 D0 06 A9 FF 8D 21
8198- 50 60 CD 03 03 D0 06 A9
81A0- 01 8D 21 50 60 C9 9B D0
81A8- 10 2C 10 C0 AD 00 C0 10
81B0- FB C9 9B F0 03 2C 10 C0
81B8- 60 C9 93 D0 14 2C 10 C0
81C0- AD 2D 50 F0 06 A9 00 8D
81C8- 2D 50 60 A9 01 8D 2D 50
81D0- 60 C9 91 D0 05 A9 03 85
81D8- 09 60 CD 12 03 D0 28 A5
81E0- FD F0 D5 AD 06 03 F0 01
81E8- 60 2C 10 C0 A9 01 8D 06
81F0- 03 A5 F9 8D 04 03 A5 FA
81F8- 8D 05 03 F8 A5 FD 38 E9
8200- 01 85 FD D8 4C C8 85 CD
8208- 13 03 D0 27 AD 06 03 F0
8210- 22 2C 10 C0 AD 04 03 8D
8218- 25 50 AD 05 03 8D 26 50
8220- A2 07 20 2B 85 A9 00 8D
8228- 06 03 F8 A5 FD 18 69 01
8230- 85 FD D8 4C C8 85 A5 FE
8238- F0 01 60 A2 00 A5 F9 8D
8240- 25 50 A5 FA 8D 26 50 20
8248- 2B 85 A5 F9 18 6D 21 50
8250- 85 F9 8D 25 50 A5 FA 8D
8258- 26 50 A2 00 20 30 81 90
8260- 0D A5 F9 38 ED 21 50 85
8268- F9 A9 00 8D 21 50 A5 FA
8270- C9 48 D0 3F A5 F9 D0 1D
8278- AD 14 03 F0 03 4C 4F 89
8280- A9 22 85 F9 A9 00 8D 22
8288- 50 8D 21 50 2C 10 C0 20
8290- 37 86 4C B3 82 C9 23 90
8298- 1A AD 14 03 F0 03 4C 4F
82A0- 89 A9 01 85 F9 A9 00 8D
82A8- 22 50 8D 21 50 2C 10 C0
82B0- 20 37 86 A5 FA 18 6D 22
82B8- 50 85 FA 8D 26 50 A5 F9
82C0- 8D 25 50 A2 00 20 30 81
82C8- 90 0D A5 FA 38 ED 22 50
82D0- 85 FA A9 00 8D 22 50 A5
82D8- FA C9 08 B0 07 A9 09 85
82E0- FA 4C EC 82 C9 88 90 04
82E8- A9 87 85 FA A5 FA 8D 26
82F0- 50 A5 F9 8D 25 50 A2 00
82F8- 4C 0A 85 A9 07 8D 27 50
8300- AC 27 50 B9 00 50 F0 11
8308- B9 00 70 8D 25 50 B9 0A
8310- 70 8D 26 50 A2 01 20 2B
8318- 85 CE 27 50 D0 E2 A9 07
8320- 8D 27 50 AC 27 50 B9 00
8328- 50 D0 06 CE 27 50 D0 F3
8330- 60 B9 00 70 18 79 0A 50
8338- 99 00 70 8D 25 50 B9 0A
8340- 70 8D 26 50 A2 01 20 30
8348- 81 90 20 AC 27 50 20 88
8350- 86 B0 18 B9 00 70 38 F9
8358- 0A 50 99 00 70 20 F9 83
8360- A9 0A 85 06 A9 1E 85 07
8368- 20 4D 86 AC 27 50 B9 00
8370- 70 D0 0B A9 01 99 00 70
8378- 20 F9 83 4C 8A 83 C9 23
8380- 90 08 A9 22 99 00 70 20
8388- F9 83 AC 27 50 B9 0A 70
8390- 18 79 14 50 99 0A 70 8D
8398- 26 50 B9 00 70 8D 25 50
83A0- A2 01 20 30 81 90 15 AC
83A8- 27 50 20 88 86 B0 0D B9
83B0- 0A 70 38 F9 14 50 99 0A
83B8- 70 20 0C 84 AC 27 50 B9
83C0- 0A 70 C9 03 B0 0B A9 87
83C8- 99 0A 70 20 42 86 4C DD
83D0- 83 C9 8F 90 08 A9 09 99
83D8- 0A 70 20 42 86 AC 27 50
83E0- B9 00 70 8D 25 50 B9 0A
83E8- 70 8D 26 50 A2 01 B9 14
83F0- 70 D0 03 20 0A 85 4C 2B
83F8- 83 B9 0A 50 C9 01 F0 06
8400- A9 01 99 0A 50 60 A9 FF
8408- 99 0A 50 60 B9 14 50 C9
8410- 07 F0 06 A9 07 99 14 50
8418- 60 A9 F8 99 14 50 60 A5
8420- FE F0 01 60 A9 07 8D 27
8428- 50 AC 27 50 B9 00 50 F0
8430- 0E B9 00 70 38 E5 F9 C9
8438- 03 90 0C C9 FE B0 08 CE
8440- 27 50 D0 E5 4C 4A 85 B9
8448- 0A 70 38 E5 FA C9 0F 90
8450- 07 C9 F0 B0 03 4C 3F 84
8458- A9 08 85 FE A9 05 8D 28
8460- 50 A5 F9 8D 25 50 A5 FA
8468- 8D 26 50 A2 00 20 2B 85
8470- A5 F9 8D 23 50 A5 FA 8D
8478- 24 50 A9 1E 8D 27 50 85
8480- 07 20 B3 85 A9 1E 85 06
8488- CE 27 50 D0 F4 60 A5 FE
8490- D0 01 60 CE 28 50 D0 09
8498- A9 03 8D 28 50 C6 FE F0
84A0- 1F A4 FE B9 01 85 AA 0A
84A8- 0A 85 07 AD 23 50 8D 25
84B0- 50 0A 85 06 AD 24 50 8D
84B8- 26 50 20 0A 85 4C B3 85
84C0- AD 23 50 8D 25 50 AD 24
84C8- 50 8D 26 50 A4 FE BE 01
84D0- 85 20 2B 85 A9 87 85 FA
84D8- 8D 26 50 A9 01 85 F9 8D
84E0- 25 50 A2 00 8E 21 50 8E
84E8- 22 50 2C 10 C0 20 0A 85
84F0- A9 00 8D 0E 03 C6 FB 10
84F8- 05 A9 01 85 09 60 4C C8
8500- 85 00 04 06 03 04 06 03
8508- 04 06 20 03 81 20 EF 80
8510- A1 02 91 00 E6 02 C8 CE
8518- 2E 50 D0 F4 AD 29 50 8D
8520- 2E 50 EE 2A 50 CE 2C 50
8528- D0 E3 60 20 03 81 20 EF
8530- 80 A9 00 91 00 C8 CE 2E
8538- 50 D0 F8 AD 29 50 8D 2E
8540- 50 EE 2A 50 CE 2C 50 D0
8548- E5 60 A9 07 8D 27 50 AC
8550- 27 50 B9 32 70 F0 0E B9
8558- 1E 70 38 E5 F9 C9 03 90
8560- 0A C9 FE B0 06 CE 27 50
8568- D0 E5 60 B9 28 70 38 E5
8570- FA C9 0F 90 07 C9 F1 B0
8578- 03 4C 65 85 B9 1E 70 8D
8580- 25 50 0A 85 07 B9 28 70
8588- 8D 26 50 85 06 A2 02 20
8590- 2B 85 AC 27 50 A9 00 99
8598- 32 70 99 1E 70 99 28 70
85A0- 20 4D 86 20 08 86 20 C8
85A8- 85 C6 FF D0 05 A9 01 8D
85B0- 14 03 60 AD 2D 50 F0 0F
85B8- A4 07 A6 06 CA D0 FD 2C
85C0- 30 C0 C6 06 88 D0 F3 60
85C8- A9 14 85 25 A9 08 85 24
85D0- 20 22 FC A5 FB 20 DA FD
85D8- A9 1E 85 24 A5 FD 20 DA
85E0- FD A9 15 85 25 A9 08 85
85E8- 24 20 22 FC AD 20 50 20
85F0- DA FD AD 1F 50 20 DA FD
85F8- AD 1E 50 20 DA FD A9 1E
8600- 85 24 A5 FC 20 DA FD 60
8608- F8 AD 1E 50 18 69 0A 8D
8610- 1E 50 90 21 AD 1F 50 18
8618- 69 01 8D 1F 50 90 16 AD
8620- 20 50 18 69 01 8D 20 50
8628- 90 0B A9 00 8D 1E 50 8D
8630- 1F 50 8D 20 50 D8 60 A9
8638- 1E 85 06 A9 C8 85 07 4C
8640- 4D 86 A9 64 85 07 A9 32
8648- 85 06 4C 4D 86 AD 2D 50
8650- F0 0D A4 07 A6 06 CA D0
8658- FD 2C 30 C0 88 D0 F5 60
8660- 00 14 0A 19 0F 1E 05 23
8668- AD 06 03 D0 01 60 AD 04
8670- 03 8D 25 50 AD 05 03 8D
8678- 26 50 A2 07 20 0A 85 A9
8680- 0F 85 06 85 07 4C B3 85
8688- AD 06 03 D0 02 18 60 B9
8690- 00 70 38 ED 04 03 C9 02
8698- 90 06 C9 FF B0 02 18 60
86A0- B9 0A 70 38 ED 05 03 C9
86A8- 0C 90 06 C9 F8 B0 02 18
86B0- 60 A9 00 8D 06 03 99 00
86B8- 50 A9 08 99 14 70 A9 05
86C0- 99 3C 70 AD 04 03 99 46
86C8- 70 AD 05 03 99 50 70 98
86D0- 48 AD 04 03 8D 25 50 AD
86D8- 05 03 8D 26 50 A2 07 20
86E0- 2B 85 A9 19 85 06 85 07
86E8- 20 4D 86 C6 06 D0 F9 68
86F0- A8 38 60 A9 07 8D 27 50
86F8- AE 27 50 BD 14 70 D0 06
8700- CE 27 50 D0 F3 60 DE 3C
8708- 70 D0 0A A9 03 9D 3C 70
8710- DE 14 70 F0 22 BC 14 70
8718- BD 46 70 8D 25 50 0A 85
8720- 06 BD 50 70 8D 26 50 B9
8728- 01 85 AA 0A 85 07 20 0A
8730- 85 20 B3 85 4C 00 87 BD
8738- 46 70 8D 25 50 BD 50 70
8740- 8D 26 50 A2 03 20 2B 85
8748- AE 27 50 A9 00 9D 00 70
8750- 9D 0A 70 20 08 86 20 08
8758- 86 4C C8 85 AD 0E 03 F0
8760- 01 60 AD 09 03 D0 27 CE
8768- 07 03 D0 21 A9 1E 8D 07
8770- 03 CE 08 03 D0 17 A9 03
8778- 8D 08 03 A9 0A 8D 0A 03
8780- A9 06 8D 0B 03 A9 01 8D
8788- 09 03 20 D5 87 60 CE 0A
8790- 03 F0 32 AD 0A 03 C9 05
8798- B0 10 A9 48 8D 26 50 A9
87A0- 13 8D 25 50 AE 0C 03 4C
87A8- 2B 85 A9 48 8D 26 50 A9
87B0- 13 8D 25 50 AE 0C 03 20
87B8- 0A 85 A9 32 85 06 A9 0A
87C0- 85 07 4C B3 85 A9 0A 8D
87C8- 0A 03 CE 0B 03 D0 05 A9
87D0- 00 8D 09 03 60 EE 0C 03
87D8- AD 0C 03 C9 0E 90 05 A9
87E0- 08 8D 0C 03 60 A9 07 8D
87E8- 27 50 AC 27 50 B9 32 70
87F0- D0 06 CE 27 50 D0 F3 60
87F8- CE 0D 03 AD 0D 03 C9 0A
8800- B0 14 B9 1E 70 8D 25 50
8808- B9 28 70 8D 26 50 A2 02
8810- 20 2B 85 4C F2 87 C9 FF
8818- F0 1D B9 1E 70 8D 25 50
8820- B9 28 70 8D 26 50 A2 02
8828- 20 0A 85 A9 14 85 06 85
8830- 07 20 B3 85 4C F2 87 A9
8838- 14 8D 0D 03 4C 16 88 AD
8840- 09 03 D0 01 60 A5 F9 C9
8848- 13 F0 01 60 A5 FA C9 48
8850- F0 01 60 A9 13 8D 25 50
8858- A9 48 8D 26 50 AE 0C 03
8860- 20 2B 85 A9 00 8D 09 03
8868- A9 1E 8D 07 03 A9 03 8D
8870- 08 03 A9 0A 8D 27 50 A9
8878- 06 85 07 A9 14 85 06 20
8880- 4D 86 C6 06 D0 F9 A9 14
8888- 85 06 CE 27 50 D0 F0 A9
8890- 01 8D 0E 03 F8 20 14 86
8898- 4C C8 85 AD 0F 03 F0 01
88A0- 60 CE 10 03 F0 32 AD 10
88A8- 03 C9 05 B0 0F A9 22 8D
88B0- 25 50 A9 87 8D 26 50 A2
88B8- 07 4C 2B 85 A9 22 8D 25
88C0- 50 A9 87 8D 26 50 A2 07
88C8- 20 0A 85 AD 10 03 0A 85
88D0- 06 A9 10 85 07 4C B3 85
88D8- A9 0F 8D 10 03 4C A1 88
88E0- AD 0F 03 F0 01 60 A5 F9
88E8- C9 22, F0 01 60 A5 FA C9
88F0- 87 F0 01 60 A5 F9 8D 25
88F8- 50 A5 FA 8D 26 50 A2 07
8900- 20 2B 85 A9 01 8D 0F 03
8908- A9 1E 85 06 85 07 A9 04
8910- 8D 27 50 20 4D 86 C6 06
8918- D0 F9 A9 1E 85 06 CE 27
8910- 8D 27 50 20 4D 86 C6 06
8918- D0 F9 A9 1E 85 06 CE 27
8920- 50 D0 F0 F8 A5 FD 18 69
8928- 01 85 FD D8 4C C8 85 AD
8930- 06 03 D0 01 60 AD 04 03
8938- 8D 25 50 AD 05 03 8D 26
8940- 50 A2 07 20 2B 85 A9 14
8948- 85 06 85 07 4C B3 85 A5
8950- F9 8D 25 50 A5 FA 8D 26
8958- 50 A2 00 20 0A 85 A9 14
8960- 8D 27 50 A9 32 85 07 85
8968- 06 20 4D 86 C6 06 D0 F9
8970- A9 1E 85 06 CE 27 50 D0
8978- F0 A9 02 85 09 60 FF
```

被困在迷宮裡的小猩猩

```
10  REM  ***************
20  REM  * BONGO DASH *
30  REM  *      BY     *
40  REM  * COMPUHOUSE *
50  REM  ***************
60  LOMEM: 17000
70  GOSUB 1000: REM INITIALIZE
80  GOSUB 2000: REM OPENING SCREE
    NS
90  GOSUB 3000: REM RE-SET GAME
99  REM  PLAY ONE GAME
100  POKE 249,1: POKE 250,135: REM
    BONGO'S START
110 LV = LV + 1: IF LV -  INT (LV
     / 5) * 5 = 0 AND RO < 8 THEN
    RO = RO + 1: POKE 251, PEEK
    (251) + 1: REM EXTRA BONGO A
    ND ROBOT
120  REM    RANDOM ROBOTS
130  FOR I = 1 TO 7:X =  INT ( RND
    (1) * 10) * 3 + 4:Y =  INT (
     RND (1) * 4 + 1): POKE 2867
    2 + I,X: POKE 28682 + I,Y *
    21 + 30: POKE 20490 + I,255:
     POKE 20500 + I,7: NEXT
140  FOR I = 1 TO 7: POKE 20480 +
    I,0: IF I <  = RO THEN  POKE
    20480 + I,1
150  NEXT
160  REM    RANDOM COCONUTS
170  FOR I = 1 TO 7: POKE 28722 +
    I,1: POKE 28702 + I,( INT ( RND
    (1) * 10) * 3) + 4:Y =  INT
    ( RND (1) * 4 + 1): POKE 287
    12 + I,Y * 21 + 30: NEXT
180  HOME : VTAB 21: PRINT " GET
    READY "
190  FOR H = 1 TO 2: FOR I = 2 TO
    7: FOR J = 1 TO I - 1
200  IF  PEEK (28702 + J) =  PEEK
    (28702 + I) AND  PEEK (28712
     + J) =  PEEK (28712 + I) THEN
     POKE 28702 + I,( INT ( RND
    (1) * 10) * 3) + 4:Y =  INT
    ( RND (1) * 4 + 1): POKE 287
    12 + I,Y * 21 + 30
210  NEXT : NEXT : NEXT : POKE 25
    5,7: REM 7 COCONUTS
230  VTAB 21: HTAB 1: PRINT "BONG
    OS :";: HTAB 20: PRINT "TREE
    S LEFT:": PRINT "SCORE  :";:
     HTAB 20: PRINT "MAZE LEVEL:
    "
240  REM    INCREASE DIFFICULTY
250  IF LV -  INT (LV / 5) * 5 =
    0 THEN  VTAB 23: HTAB 1: PRINT
    "WATCH OUT! ANOTHER ROBOT CA
    ME IN!": FOR I = 1 TO 10: POKE
    T,I * 25: POKE L,50: CALL SO
    UND: NEXT : FOR I = 1 TO 150
    0: NEXT : VTAB 23: HTAB 1: CALL
     - 868
260  POKE KB,0: CALL 32768: REM P
    LAY ONE ROUND
270 I =  PEEK (9): IF I = 1 THEN
    330: REM YOU LOSE
280  IF I = 3 THEN 370: REM YOU Q
    UIT
290  REM    YOU WIN
300  FOR I = 1 TO 10: FOR J = 20 TO
    1 STEP  - 1: POKE T,I + J: POKE
    L,J: CALL SOUND: NEXT : NEXT

310  HOME : INVERSE : VTAB 21: HTAB
    5: PRINT " NEW MAZE...GOOD L
    UCK BONGO! ": NORMAL : GOSUB
    500: GOTO 100
320  REM    YOU LOSE
330  FOR I = 1 TO 255 STEP 10: POKE
    T,I: POKE L,256 - I: CALL SO
    UND: NEXT
340  HOME : VTAB 21: PRINT "BYE B
    YE BONGO!  YOU'RE A DEAD GOR
    ILLA!": PRINT : INVERSE : PRINT
    " HIT Q TO QUIT, ANY OTHER K
    EY TO PLAY ";: NORMAL
350  POKE KB,0: GET A$: IF A$ <  >
    "Q" THEN  GOSUB 500: GOTO 80
360  REM    YOU QUIT
370  HOME : VTAB 21: HTAB 8: PRINT
    "SEE YOU NEXT TIME BONGO! ";

380  FOR I = 1 TO 3: FOR J = 1 TO
    10: FOR K = 1 TO 5: POKE T,J
    : POKE L,K: CALL SOUND: NEXT
    : NEXT : POKE T,I * 3: CALL
    SOUND: NEXT
390  FOR K = 1 TO 1000: NEXT : TEXT
    : HOME : POKE KB,0: END
499  REM  DRAW MAZE
500  HGR
510 WL = 21: HCOLOR=  INT ( RND (
    1) * 7 + 1): IF  PEEK (228) =
    128 THEN 510
520  FOR I = WL + 2 TO 255 - WL STEP
    WL: FOR J = WL + 2 TO 150 -
    WL STEP WL:WA =  INT ( RND (
    1) * 2 + 1): IF WA = 1 THEN
     HPLOT I,J TO I + WL,J: HPLOT
    I,J + 1 TO I + WL,J + 1
530  IF WA = 2 THEN  HPLOT I,J TO
    I,J + WL: HPLOT I + 1,J TO I
     + 1,J + WL
540  POKE T,J: POKE L,6: CALL SOU
    ND: NEXT : NEXT
550  HPLOT 0,4 TO 255,4 TO 255,14
    9 TO 0,149 TO 0,4: HPLOT 1,5
     TO 254,5 TO 254,148 TO 1,14
    8 TO 1,5: IF  PEEK (228) < 1
    28 THEN  HCOLOR= 0: GOTO 570

560  HCOLOR= 4
570  HPLOT 130,4 TO 148,4: HPLOT
    130,5 TO 148,5: HPLOT 0,72 TO
    0,82: HPLOT 1,72 TO 1,82: HPLOT
    130,148 TO 148,148: HPLOT 13
    0,149 TO 148,149
580  HPLOT 254,72 TO 254,82: HPLOT
    255,72 TO 255,82: IF LV < 9 THEN
     RETURN
590  HPLOT 233,130 TO 233,147: HPLOT
    234,130 TO 234,147: FOR I =
    130 TO 148: HPLOT I,65 TO I,
    87: NEXT
600  FOR I = 67 TO 85: HPLOT 128,
    I TO 150,I: NEXT : RETURN
999  REM  INITIALIZE
1000  HGR : TEXT : HOME
```

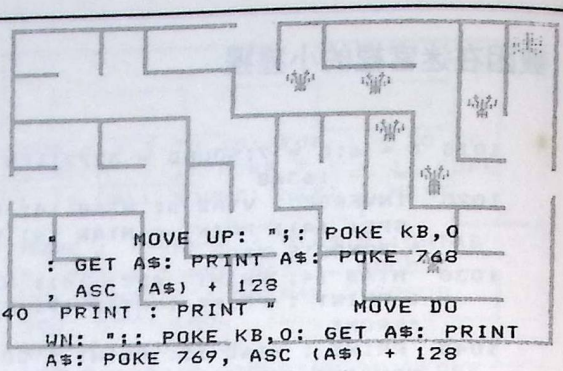
被困在迷宮裡的小猩猩

```
1010 T = 6:L = 7:SOUND = 32771:KB
     = - 16368
1020  INVERSE : VTAB 6: HTAB 14: PRINT
     SPC( 14): PRINT : HTAB 14: PRINT
     " BONGO'S BASH "
1030  HTAB 14: PRINT  SPC( 14): NORMAL
     : PRINT : PRINT : HTAB 19: PRINT
     "FROM"
1040  PRINT : HTAB 13: PRINT " CO
     MPUTING AGE"
1050  VTAB 22: HTAB 13: FLASH : PRINT
     " GETTING  READY ": NORMAL
1070  PRINT  CHR$ (4);"BLOAD BONG
     O.SHAPES"
1080  PRINT  CHR$ (4);"BLOAD BONG
     O.OBJ"
1090  REM     CREATE LOOKUP TABLE
1100 LOC = 24576:CO = 0: FOR H =
     0 TO 2:BYTE = 31: FOR I = 0 TO
     3:BYTE = BYTE + 1: FOR J = 0
      TO 1: FOR K = 0 TO 7: POKE
     LOC + CO,BYTE + (4 * K):CO =
     CO + 1: NEXT : NEXT : NEXT :
      NEXT
1110  POKE LOC + CO,255:CO = CO +
     1: IF CO < 256 THEN 1110
1120 BYTE = 0: GOSUB 1150:BYTE =
     40: GOSUB 1150:BYTE = 80: GOSUB
     1150
1130  POKE LOC + CO,255:CO = CO +
     1: IF CO < 512 THEN 1130
1140  POKE 20525,1: GOTO 1160
1150  FOR I = 0 TO 3: FOR J = 0 TO
     1: FOR K = 0 TO 7: POKE LOC +
     CO,BYTE:CO = CO + 1: NEXT :B
     YTE = BYTE + 128: NEXT :BYTE
      = BYTE - 256: NEXT : RETURN

1160  VTAB 22: HTAB 9: PRINT "PRE
     SS ANY KEY TO BEGIN ";: POKE
     KB,0
1170 WA =  INT ( RND (1) * 2 + 1)
     : IF  PEEK ( - 16384) < 128 THEN
     1170
1180  HOME : RETURN
1999  REM  SET GAME VALUES
2000  GOSUB 500: REM DRAW MAZE
2010  HOME : VTAB 21: HTAB 1: INVERSE
     : PRINT ":                  :": PRINT
     "  BONGO'S BASH  ": PRINT ":
                    :": NORMAL : VTAB
     21: HTAB 18: PRINT "BY JOHN
     & RALPH ROMERO"
2020  VTAB 23: HTAB 21: INVERSE :
      PRINT " PRESS ANY KEY ";: NORMAL
     : POKE KB,0: GET A$
2030  REM     OPENING MENU
2040  HOME : VTAB 21: PRINT  SPC(
     8);"<SPACE>   INSTRUCTIONS":
      PRINT  SPC( 8);"<RETURN>  P
     LAY GAME": PRINT  SPC( 10);"
     <N>     FOR NEW MAZE"
2050  VTAB 1: POKE KB,0: GET A$
2060  IF A$ =  CHR$ (13) THEN  RETURN

2070  IF A$ = "N" THEN  GOSUB 500

2080  IF A$ = " " THEN  GOSUB 400
     0
2090  GOTO 2040
```

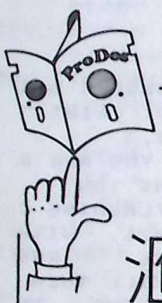


```
2999  REM  SET GAME VALUES
3000  HOME : VTAB 21: PRINT "DIFF
     ICULTY 1-6 (6 = EASIEST) 3";
     : HTAB 30: POKE KB,0: GET A$

3010  IF A$ =  CHR$ (13) THEN A$ =
     "3"
3020  IF A$ < "1" OR A$ > "6" THEN
     3000
3030  POKE 8,25 *  VAL (A$) + 75:
      REM SET SPEED
3040  POKE 768,193: POKE 769,218:
      POKE 770,136: POKE 771,149:
      POKE 786,160: POKE 787,141:
      REM DEFAULT CONTROLS
3050  HOME : VTAB 21: PRINT "CHAN
     GE CONTROLS? (Y/N) N";: HTAB
     24: POKE KB,0: GET A$: IF A$
      = "Y" THEN  GOSUB 5000
3060  HOME : VTAB 21: PRINT "STAR
     T WITH HOW MANY ROBOTS? (1-7
     ) 1";: HTAB 35: POKE KB,0: GET
     A$
3070  IF A$ =  CHR$ (13) THEN A$ =
     "1"
3080  IF A$ < "1" OR A$ > "7" THEN
     3060
3090 RO =  VAL (A$)
3100  POKE 251,3: REM 3 BONGOS
3110 LV = 0: POKE 252,0: REM LEVE
     L 0
3120  POKE 253,5: REM 5 TREES
3140  POKE 780,7: REM SHAPE OF FI
     RST BONUS FRUIT
3150  POKE 20510,0: POKE 20511,0:
      POKE 20512,0: REM ZERO SCOR
     E
3160  RETURN
3999  REM  INSTRUCTIONS
4000  TEXT : HOME
4010  HTAB 14: PRINT "BONGO'S BAS
     H": FOR I = 1 TO 40: PRINT "
     -";: NEXT : POKE 34,3
4020  PRINT : PRINT "   YOU ARE B
     ONGO THE GORILLA AND YOU
      MUST EAT ALL THE FLASHING C
     OCONUTS IN   THE MAZE.  WITH
      YOU IN THE MAZE ARE     ROB
     OT SENTRIES.  YOU KILL THEM
     BY       DROPPING TREES.  TH
     EY KILL YOU BY       TOUCHIN
     G YOU."
4030  PRINT : PRINT "   IN THE LO
     WER-RIGHT CORNER OF THE
      SCREEN IS ANOTHER TREE YOU
     CAN GET. IN  THE MIDDLE OF T
     HE MAZE ARE FLASHING     PRI
     ZES. YOU GET EXTRA POINTS FO
     R THESE.": GOSUB 4090
4040  PRINT "   THE TWO SIDE DOOR
     S ARE YOURS AND THE  TOP AND
      BOTTOM DOORS ARE THE ROBOTS
          YOU CAN'T GO THROUGH TH
     EIRS AND YOURS   ARE BLOCKED
      FOR THEM."
4050  PRINT : PRINT "   YOU BEGIN
      WITH ONE ROBOT IN THE MAZE,
      AND A MEDIUM SPEED OF 3. YO
     U CAN CHANGE THESE SETTINGS
     OR THE GAME CONTROLS."
```

被困在迷宮裡的小猩猩



```
4060  PRINT : PRINT " HERE ARE TH
      E PRESENT GAME CONTROLS:": PRINT
      "        A--UP          <-
        ->": PRINT "        Z--DOWN
            RIGHT    LEFT"
4070  PRINT : PRINT "        <SPAC
      E>    DROP TREE": PRINT "
         <RETURN>  RECOVER TREE": PRINT
      "        <CTRL>-Q   QUIT GAME"
      : PRINT "        <CTRL>-S   TO
      GGLE SOUND": PRINT "        <
      ESC>      PAUSE GAME"
4080  GOSUB 4090: TEXT : POKE  -
      16304,0: POKE  - 16297,0: RETURN

4090  VTAB 23: HTAB 12: INVERSE :
      PRINT " PRESS ANY KEY ";: NORMAL
      : POKE KB,0: GET A$: HOME : RETURN
4999  REM  CHANGE CONTROLS
5000  TEXT : HOME
5010  HTAB 14: PRINT "BONGO'S BAS
      H": FOR I = 1 TO 40: PRINT "
      -";: NEXT : POKE 34,2
5020  PRINT : PRINT " THE CONTROL
      S ARE: ";:KY = 768: GOSUB 51
      00: PRINT ",";:KY = 769: GOSUB
      5100: PRINT ",";:KY = 770: GOSUB
      5100: PRINT ",";:KY = 771: GOSUB
      5100
5030  PRINT : PRINT : PRINT : PRINT
      "      MOVE UP: ";: POKE KB,0
      : GET A$: PRINT A$: POKE 768
      , ASC (A$) + 128
5040  PRINT : PRINT "      MOVE DO
      WN: ";: POKE KB,0: GET A$: PRINT
      A$: POKE 769, ASC (A$) + 128

5050  PRINT : PRINT "      MOVE LE
      FT: ";: POKE KB,0: GET A$: PRINT
      A$: POKE 770, ASC (A$) + 128

5060  PRINT : PRINT "      MOVE RI
      GHT: ";: POKE KB,0: GET A$: PRINT
      A$: POKE 771, ASC (A$) + 128

5070  PRINT : PRINT "      PLANT T
      REE: ";: POKE KB,0: GET A$: PRINT
      A$: POKE 786, ASC (A$) + 128

5080  PRINT : PRINT "      TAKE TR
      EE: ";: POKE KB,0: GET A$: PRINT
      A$: POKE 787, ASC (A$) + 128

5090  TEXT : POKE  - 16304,0: POKE
       - 16297,0: RETURN
5100 A$ =  CHR$ ( PEEK (KY)): IF
      PEEK (KY) = 136 THEN A$ = "
      <-"
5110  IF  PEEK (KY) = 149 THEN A$
      = "->"
5120  PRINT " ";A$;: RETURN
```



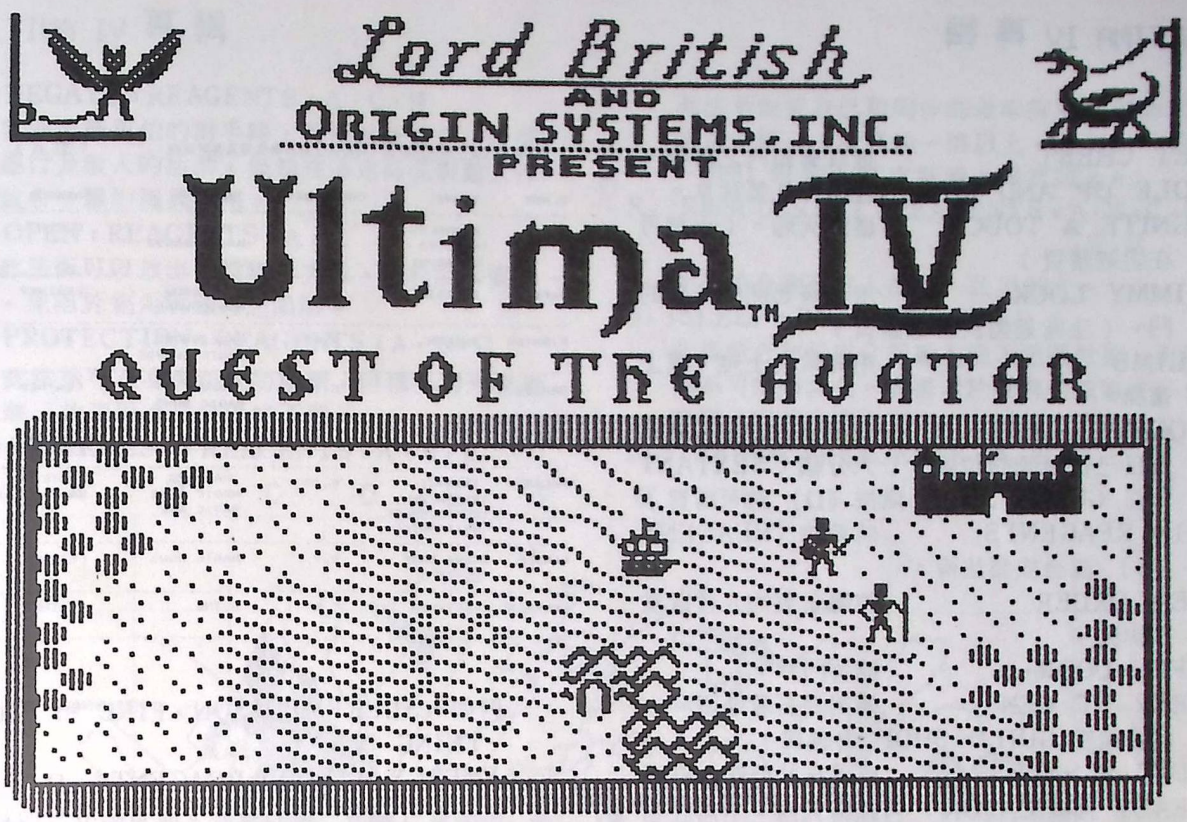
滙編語言 編譯器 的使用方法

APPLE公司的PRODOS ASSEMBLER TOOL KIT中有一個叫做EDASM的系統程式，它是PRODOS的組合語言編譯器，雖然目前並沒有使用指南，但事實上它的使用方法是很容易的，從立即模式中進入只要鍵入 EDASM·SYSTEM（RET）即能進入它的EDITOR（編輯器），這時你可以像使用DOS TOOL KIT 般使用一切編輯模式命令來編寫，修改，除錯組合語言的原程式，由於兩者之間並無甚麼大分別，因此指令都大致相同。

當你的程式完成後，第一件事要先把它貯回磁碟中，指令格式是SAVE／VOLUME NAME／PATHNAME ，以後再次修改或編輯時再次載入只要 LOAD／VOLUME NAME／PATHNAME 即可，當你要組譯之時，首先要把記憶體內的原程式先貯存到磁碟上，再以NEW 指令清除緩衝器，鍵入如下指令

ASM／VOLUME NAME／PATHNAME，／VOLUME NAME／OBJECT PATHNAME

編譯器便會把磁碟上的原程式組譯，完成後磁碟上便會多了一個目的程式檔了，不過注意不要貼上WRITE PROTECT到磁碟上，否則會引致WRITE ERROR。另外在使用ASM指令之前，要先用NEW 指令，否則會引致BUFFER ERR－OR ，無法組譯。



ULTIMA IV的玩法

撰稿：先鋒電腦軟件公司

像電影「魔域仙踪」一類的魔幻境界歷險遊戲中，以 ULTIMA 系列最爲出色。ULTIMA 系列均擁有精彩懸疑的故事架構，細緻而充滿動感的畫面構圖。近期的版本更能配合 MOCKING BOAR-D 隨着情節的發展襯托上緊張的立體聲背景音響效果，所以能使遊戲者全神投入故事角色中，達到心景交融的境界。

ULTIMA IV 故事敍述 LORD BRITISH 的帝國遭受到魔王「CODEX」日漸擴張的邪惡勢力威脅。爲着保衞國土及使平民有安定的生活，L-ORD BRITISH 從國境內挑選了八名精悍善戰的勇士對抗魔王的襲擊。遊戲者需要肩負這個重任，離開自己的家鄉，沿途尋找另外的七名戰士，聯手對抗魔王。

不過，戰士必須能通過嚴格的考驗，鍛練到成爲世上性格最完美，品德最高尚的人，才有資格消滅魔王。達到完美的標準乃根據戰士的所作所爲而定。共有八項先決條件分別爲：謙遜（HUMILIT-Y）、勇敢（VALOR）、誠實（HONESTY）、同情心（COMPASSION）、公正（JUSTICE）、名譽（HONOR）、犧牲（SACRIFICE)與及理靈（SPIRITUALITY)。要達到各項的要求，必須緊記在 COMBAT 狀態時切勿逃走，以免被視爲懦弱畏縮。在城內購物時，要如數付款，勿生貪念才稱得上誠實。犧牲精神方面可多嘗試捐血給「H-EAL」的店鋪（最少三次以上）。要有同情心，布施給沿途遇上的乞丐。在遇到寶箱時必須在伙伴中輪流開啓，以示公正無偏。而在城市內不可亂動路旁的寶物或屠殺居民，否則會嚴重影響名譽。還有要做到謙讓有禮，與城市裏的人交談時，切不可一時說對，一時又說不對。此外，戰士們還需找齊八粒聖石及其他重要寶物，例如：BELL，BOOK 及 CANDLE 等，將寶物混合成一條鎖匙，方能進入魔王 CODEX 的巢穴。

說完遊戲的背景及目的後，現在應是教導各位怎樣玩本遊戲，以下是遊戲的控制鍵說明：

A) TTACK　攻擊敵人，使用時須註明方向。
B) OARD　登上戰艦或馬匹。
C) AST　施用法術。
D) ESCEND　用繩索或梯級向下爬。
E) NTER　進入城市、堡壘、地牢或鄉村等。
F) IRE　在戰艦上向敵人開炮轟擊。（只能向船側方向發射）

G)ET CHEST　提取寶箱內之財物。
H)OLE UP AND CAMP　在城外札營休息。
I)GNITE A TOUCH　燃點火炮。（火把可在店鋪購買）
J)IMMY LOCK　用法術匙開啓特別的門。（法術匙需於店鋪購買）
K)LIMB　用繩索向上爬，或上樓梯。
L)OCATE POSITION　使用「六分儀」來確定自己在地圖的位置。（六分儀〔SEXTANT〕在 GUILD SHOP 處按（D）鍵可購買）
M)IX REAGENTS　將葯物（REAGENT-S）混合成爲法術。
N)EW ORDER　將隊伍裏的人位置重新排列。
O)PEN DOOR　開啓門戶。
P)EER AT GEN　使用鑽石看地圖。（鑽石需在 GUILD SHOP 處購買）
Q)UIT & SAVE　將現在狀態貯入磁碟
R)EADY A WEAPON　預備武器。（需註明武器名稱）
S)EARCH　搜索物件。
T)ALK　在城內與人交談。
U)SE　用搜索所得到的物件。
V)OLUME　將聲音系統開或關。
W)EAR ARMOUR　穿戴盔甲。
X)IT　落馬或船。
Y)ELL　控制賽馬速度。
Z)TATS　查看各戰士及遊戲一切資料。
"←"　向右移。
"→"　向左移。
"RETURN"　向上移。
"/"　向下移。

以下是魔法（MAGIC SPELLS）簡介：

A.)AWAKEN:REAGENTS:B,C
施用法術可將對方喚醒，無論該人是在休息時被襲或被法術弄暈，均可使用。

B.)BLINK:REAGENTS:D,E
此法術用以幫助你的 PARTY 穿越一些平原、高山，甚至河流。但距離只限制在八格範圍內，且於使用時要註明方向。

C.)CURE: REAGENTS:B,C
用以中和毒性之法術。

D.)DISPEL:REAGENTS:A,C,F
此法術可以化解任何的 ENERGY FIELD，例如：SLEEP，POISON，FIRE 和 LIGHTNING 等。

E.)ENERGY FIELD: REAGENTS:D,F
此法術可製造一個能量牆。能量牆分爲四種：SLEEP，POISON，FIRE 和 LIGHTNING。作用爲防止其他人侵入或接近之防衛性的武器。

F.)FIRE BALL:REAGENTS:A,F
在使用人手中可形製成一團火球直接向敵人擊射，其威力驚人。

G.)GATE TRAVEL:REAGENTS:A,H,F
此法術有 MOON GATE 的功能，直接到達所想到的 MOON GATE。（以號碼表示）

H.)HEAL:REAGENTS:B,D
用以幫助 PARTY 中任何一人治療傷勢。

I.)ICE BALL:REAGENTS:F,H
此法術的攻擊力非常驚人，甚至爲 FIRE B-ALL 之上，但可惜 REAGENTS H 十分罕有，並不容易尋到。

J.)JINX:REAGENTS:F,G,H
施行此法術可以令敵人自相殘殺。

K.)KILL:REAGENTS:F,G
此乃最具攻擊力的法術，可將敵人立刻置於死地。

L.)LIGHT:REAGENTS:A
這法術代替 TORCH 的功用，在黑暗的 DUNGEONS 放出光明。

M.)MAGIC MISSILE:REAGENTS:A,F
此法術是一種較輕殺傷力的攻擊武器，因其所需 MAGIC POINTS 亦較少，故適合小型戰況。

表A是八戰士出現之座標，可配用之最高級武器及盔甲裝備。（表A）

CLASS	TOWNE	COORDINATES	WEAPONY	ARMOUR
MAGE	MOONGLOW (NEAR TO LYCAEUM)	K,7	STAFF MAGIC WANDS	CLOTH
BARD	BRITAIN	D,6	CROSSBOW MAGIC WANDS	LEATHER
FIGHTER	JHELOM	B,12	ALL SWORD MAGIC SWORDS	ALL
DRUIDS	YEW	C,3	MAGE MAGIC BOWS	LEATHER
TINKER	MINOC	H,2	AXE	LEATHER CHAIN MAIL
PALADIN	TRINSIC (SOUTHERN (SOUTHERN OF BRITANNIA)	E,10	HALBERD MAGIC BOWS MAGIC AXE	PLATE OR MAGIC PLATE
RANGER	SKARA BRAE	B,7	MAGIC BOWS	LEATHER
SHEPHERD	MAGINCIA	T,9	SLING	LEATHER

N.)NEGATE：REAGENTS：A，C，H
當遇上極可怕的對手時，可用此法術停止時間運行及敵人的活動，無論在逃走時或對敵人作致命一擊均極具作用。

O.)OPEN：REAGENTS：A，E
此法術可以放出一股無形力量，用於開啓寶箱，免陷於箱內所藏有之陷阱。

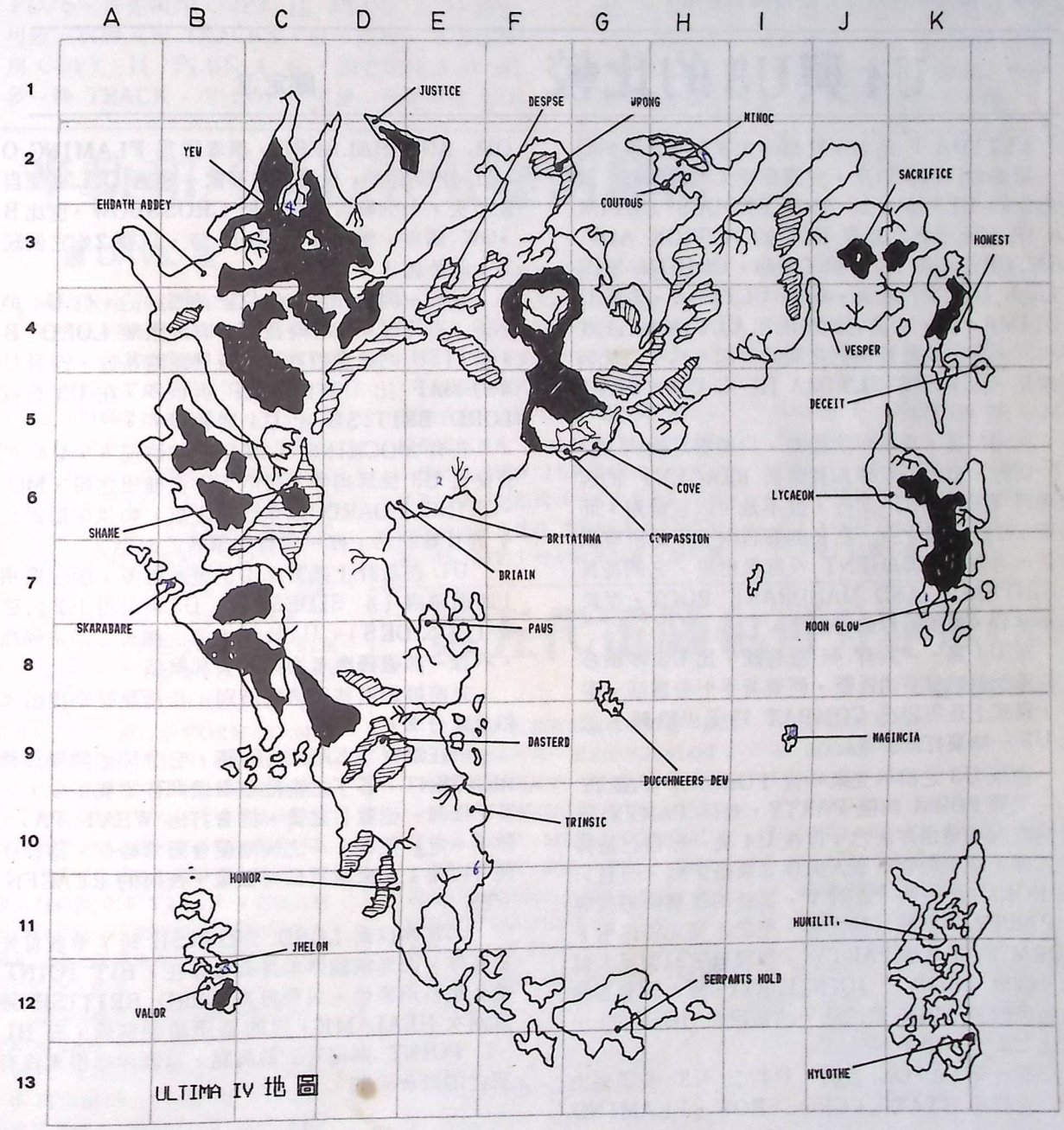
P.)PROTECTION：REAGENTS：A，B，C
此法術可在身旁組成防護網，阻擋一切外來攻擊，效用雖大，但維持不長。

Q.)QUICKNESS：REAGENTS：A，B，E
此法術能使自己和同伴的效率提高，例如攻擊或逃走時，速度可快一倍以上，並能在一 RO -UND 內每人可攻擊敵人兩次以上。

R.)RESCURRECT：REAGENTS：A，B，C，D，E，H
此術能令剛死之人復生，效用頗大。

S.)SLEEP：REAGENTS：B，D
此術能令部份或全部敵人陷入睡眠狀態，但時間不可維持太久。受害者於就算被攻擊亦不會甦醒，直至在夢中死去。

T.)TREMOR：REAGENTS：A，E，H



ULTIMA IV 地圖

此乃攻擊力甚強之法術，能使敵人的所在地產生瞬息的強烈地震。能迅速殲滅敵人。

U.)UNDEAD：REAGENTS：A，C
用以驅走 UNDEAD MONSTERS。

V.)VIEW：REAGENTS：G，H
此術功能近似 PEER AT GEMS，能看清用術者所處地方的周圍環境。

W.)WIND CHANGE：REAGENTS：A，E
此術能將周圍的風向隨用術者的意願改變，在乘船出海或乘坐汽球升空時甚爲有用。

X.)XIT：REAGENTS：A，D，E
當在 DUNGEON 遇險時，可用此術逃生，直抵洞口。

Y.)UP：REAGENTS：D，E

Z.)DOWN：REAGENTS：D，E
在 DUNGEON 時，可憑Y或Z兩個法術隨意上落一層。

表A是八戰士出現之座標及可配用之高級武器及盔甲裝備。■

U4與U3的比較

鄺定遠

ULTIMA I 這名字我相信大家都不會感到陌生，這遊戲自面世以來，一直都受大家的歡迎，及後推出的 ULTIMA II AND EXODUS ULTIM-A III，更成爲當時最 HIT 的 ACTION ADV-ENTURE GAME，隔了多時，ORIGIN SYS-TEM INC又推出新一輯的 ULTIMA，名爲U-LTIMA IV，由於它的玩法和 ULTIMA III 並不相同，所以我談談他們之間的分別。爲了方便稱呼起見，以下簡稱 ULTIMA III 爲 U3，ULTI-MA IV 爲 U4。

在 U3 裏，你想用法術時，只要有足夠的 M.-P 便行，但在 U4 你却要到賣 REAGENT 的店舖購買 REAGENT 才行，但不是可以立刻用，而是要 MIX 好後才行，在R的商店內，不是所有俱齊全，有兩種 REAGENT 沒有得出售，它們是N-IGHTHADE AND MANDRAKE ROOT，至於拿不拿到，便要看你的運氣了。

在 U4 裏，一共有 44 隻怪獸，比 U3 多很多，你要應付這樣多的怪獸，便要身手十分靈活，幸好，程式上是可以在 COMBAT 時退出戰陣，這比 U3 一味要打好得多。

在玩 U3 之前，先來一個 FORM A PARTY，一共要 FORM 四個 PARTY，每個 PARTY 屬於何許人也是由你決定；但在 U4 裏，你自己是甚麼人種，便要由吉卜賽人與你算命後才知，而且只是 FORM 你一個 PARTY，其餘的便要慢慢找尋，PARTY 一共有八個之多，至於怎樣知道你所F-ORM 的是甚麼 PARTY，很簡單，只要見人就問，問他 JOIN 不 JOIN PARTY，那人便會答你，如果他想 JOIN 你的話，他是自動 JOIN 的，這比 U3 進步得多。

至於 WEAPON 方面，U4 比 U3 多了些出來，它們是 STAFF，CROSSBOW，FLAMING OIL AND HALBERD，筆者認爲 FLAMING O-IL 比較厲害，只要那些怪獸一經過 OIL 後便自動消失，十分厲害，而那把 CROSSBOW，也比B-OW 厲害，對待 PYTHON 時，只要 2炮它便玩完，花款確比 U3 多得多。

U3 一開始你身在的位置是固定的，但 U4 則不是，有時在東，有時在西，有時距離 LORD B-RITISH 的城堡近些，有時則距離甚遠，而且U4的 MAP 比 U3 的 MAP 大很多，在 U3 裏找 LORD BRITISH 比 U4 容易很多。

在有 MOCKING BOARD 的情況下，U4 的音樂比 U3 悅耳得多，可能 U3 在推出之時，MO-CKING BOARD 還未推出之故，如果你聽過U4 的音樂的話，你一定贊成筆者的說話。

U4 在設計上確實比 U3 複雜得多，所以要用上四面磁碟（4 SIDES），而 U3則只用上2面磁碟（2 SIDES）。U4比較上適合擁有二部磁碟機，不像一部磁碟機者換來換去那麼煩。

先前提過有些事情較特別，在這裏特別提出來和大家分享。

㈠在購買 REAGENT 時，它會問你需要那種 REAGENT，答了之後他會繼續問你需要多少，你答了任何一個數目之後，他會打出 WHAT PAY，你『一定』要答 1，之後他便會照樣給你，這樣你便可以用1元來購買任何數量、任何的 REAGEN-TS 了。

㈡當你找到 LORD BRITISH 時，筆者首先恭喜你，因爲無論你本身有毒存在，HIT POINT 是0也不用慌忙，只要對着 LORD BRITISH 連說兩次 HEAL ME，你的毒跟着便解除，而 HI-T POINT 則提升到最高點，這表示你根本沒有死亡的機會。■

如何抄錄U4

■ 林智勤

ORIGIN SYSTEM 繼成名作 ULTIMA II 及 III 後，LORD BRITISH 乘勝追擊，推出U-LTIMA IV ——QUEST OF THE AVATAR。近來在高登已可買到，但以筆者所查到，只有三盒原裝到了香港，筆者便花了數十元去 COPY ，但物有所值，四面碟加一份 MANUAL 有齊 SPE-LL，MAP 等。回家後便找尋 COPY 方法，爲了找出最簡便及最準確的方法，筆者便看看其 AD-DRESS MARK (AM)，發覺每個 AM 都不同，筆者便想到用對付 AM不同而又最快的 COPY][PLUS。首先我用 COPY][PLUS 3.0，因它可以一次讀五條 TRACKS，但不成功，筆者便改用 COPY][PLUS 4.4C，因它能比 5.0 讀多一條 TRACK，但 COPY 完後，發覺不能 BO-OT 起，我便用 E.D.D III 補 COPY TRAC-K $0，再 BOOT 機，成功了，這只要 COP-Y ULTIMA IV 中的 PROGRAM 一面，其他三面如 UNDER WORLD 等可用一般 COPY A的 COPY PROGRAM 便可。現以簡述 COPY 方法如下：

㈠以 COPY][4.4 抄 $0—$22。

㈡以 E.D.D III 抄 $0

這便可和朋友分享你的 ULTIMA IV 了。補充一提的是在使用 COPY][PLUS時將參數值 31 之 01 改爲 00 是會令資料比較完整，對抄U-LTIMA IV 更有幫助，如大家不知如何使用C-OPY][PLUS 或修改參數，筆者已在第二十期中刊登使用方法，大家可參考參考。■

半軌和¼軌LOCK碟的秘密

■ DAVID 唐

現在玩的套裝遊戲軟件，我們會發現，它們很多時有TRACK0·5及T 0.25 的出現，如 ARCHOW 及 KARATE — KA便是。T 0.5及T 0.25 眞的能夠寫上資料嗎？如果是眞的，一片 5.25 吋 DISK不就成了 160 TRACKS 嗎？我們知道一片磁碟（ 5.25 吋）當然沒有160 TRACKS，因此半及¼軌也不能寫上東西，那爲甚麼抄碟時又經常接觸到這類的半軌和¼軌呢？

在我告訴大家這個秘密之前，你是否有留意到如果是 5.25 TRACK（只是一個例子）的出現時，5 TRACK 是不會寫資料的，很明顯T 5，T5.25，T 5.5 及T 5.75 其實都是同一條 TRACK，不過只是 FORMAT 時有點改變或偏差罷了！ NIBBLE AWA—YII 的 AUTO—HALF便是這個原理，你現試用它來抄一片有半軌的磁碟，在正常的情況下，它不會去理會半軌的，但當它發覺軌上資料有古怪時，便會去自動抄 0.5 TRACK ，它就是利用這個原理了。那麼到底它是怎樣做出來和怎樣把這些軌應用在 LOCK DI—CK 之上呢？

這些半軌和¼軌是可用一些專用 PROGRAM來 FORMAT的，FOR—MAT 後同軌是不能寫資料的，就是在 FORMAT T5.25 後；T5便不會寫東西了。既然半軌及¼軌不能寫東西，那麼要來有甚麼用呢？當然是防止「盜錄」用。方法是這樣的：

首先要利用 TRACK 0

在 TRACK 0 寫上一程式

這個副程式是用來追查特別軌的，即是半軌，¼軌。每當 BOOT 碟後，這程式都首先 LOAD入 MEMORY，然後程式便會 JSR $ XX（ 20 XX）跳至副程式（ JSR $XX）這個指令通常也放在 TRACK 0），然後查一些「特定」的半軌及¼軌，如果成功的話 （查到有指定的半軌或¼軌），便 RTS（60），否則便要用其他的處理方法，如： HANG 機，BOOT 碟等等。這樣大家應了解了吧！

如果把這些特別軌去除，可查 TR—ACK 0 ，遇到有 20 XX 的，便改 EA EA EA ，記着，這個指令只會出現一次，所以，要看清楚那一個，如果你不懂 ASSEMBLY，最好不要亂改了，改錯了，把眞正有用程式改到亂七八糟，以致不能玩了。■

第五代電腦也可分享MCS

■ NIBBLE

相信大家都有過這樣的經驗，就是自己的電腦內有一張Z80 咭在SLOT 4，而嘗試去BOOT 一些MOCKING BO—ARD 版本的GAME 或 UTILITY ，往往是失敗的，MUSIC CONSTRUC—TION SET 就是一個例子，當然，最簡單的方法就是將Z80 咭移走，但是如果你的電腦是BUILT—IN Z80 ，這個方法是絕不可能的。今日，筆者就爲大家提供一個方法去改MCS ，令大家也可享用它。

方法：

1. BOOT 一隻有DOS 的磁碟。（注意：不要BOOT MCS）
2. 進入BASIC狀態，放入MCS碟
3. UNLOCK A4
4. BLOAD A4
5. CALL—151
6. 7FED：EA EA EA
7. 7F00：EA EA EA EA EA EA
8. 8CCC：EA EA EA
9. 8002：EA EA EA
10. 8028：EA EA EA
11. 按<CTRL—C>去BASIC
12. BSAVE A4，A$4A00，L$4B60 成功！！

注意：BOOT了MCS之後，請不要選擇用MOCKING BOARD 來做輸出，這對於BUILT—IN Z 80 電腦的朋友，也是不行的。■

WINTER GAMES

軟件名稱：WINTER GAMES
軟件類型：運動比賽
評　　分：＊＊＊

上期介紹過SUMMER GAMES 2，今期再爲讀者介紹WINTER GAM－ES。

WINTER GAMES 製作上來說已不俗，但由於玩者對那些比賽的項目似乎不大熟識關係，所以在控制方面較難掌握。

經過筆者悉心研究後，總算領會到其中的奧妙，現分析每個LEVEL 的玩法。

1.HOT DOG

在螢幕上，玩者會發現有一傾斜的山坡，玩者要用最好看的方式，從山頂跳下來，電腦分別提供了六種花式給你選擇，包括

前傾	U	O	後傾
後翻	Y	K	前翻
脚傾	N	，	前身傾

玩法秘訣：當比賽者跳起時，立刻按Y，當就快到達地面時，再按O，如無意外，你應該取到非常高的成績。

2.BOBSLED

這項目是要玩者控制一架雪車在雪坑內飛馳。當開始時按BUTTON 0 ，跟着把控桿左右擺動，以保持平衡。筆者在這項目僥倖完成整條路程，相信讀者也能夠做得到吧。

3.FIGURE SKATING

這是一項花式溜冰比賽，玩者要竭盡所能才能拿到好成績，遊戲提供了多種花式給玩者，分別如下：

U	轉體跳	K	前踏
I	燕子式旋轉	N	直立式跳
O	前轉體跳	M	俯身旋轉
J	後踏	，	前直立式跳

雖然有那麼多花式給玩者使用，但每一種花式也有些限制，例如U、N、I、M 等花式，只限於後踏時才可用，而O 、，等花式，只限於前踏時使用，如果那些花式是跳起的話，一定要在完成動作後按J 平衡。

4.FREE SKATING

FREE SKATING的玩法大致上和FIGURE SKATING 大致一樣，只是時間方面不同罷了。

5.SPEED SKATING

這項目是這遊戲最易玩的項目，玩者只需開始時不停地按J及K或用JOY－STICK 左右擺動便可。

6.SKI JUMP

這項目是最令玩者傷腦筋的。開始時，比賽者會從滑雪高台中滑落，按SPACE BAR跳起，跟着在螢幕右上方便會看到比賽者，此時立刻按K把身子向前傾，大約待二秒後，再按J 把身子向後拉，再按I 穩定位置，此時你應成功降落而取得非常高分了。

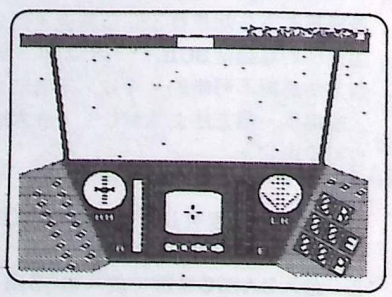
7.BIATHLON

這項目是要考玩者的反應及速度，當比賽開始時，玩者應把搖桿作左右擺動，當看見斜坡時，把控桿拉落作俯衝，當上斜坡的時候，把控桿快速地左右擺動，如是者經過兩圈後，你便會看見一塊板有五個目標，此時你應把控桿拉後上彈，再拉前準備發射，當目標接近白點時，立即按BUTTON 0 發射，那麼白點便會變成黑點，亦即表示玩者得到一分了。

總括來說，冬季奧運會已比前兩個奧運會的製作上較好，而且更加上八國評判一齊計分，令這遊戲生色不少，相信不久便會推出WINTER GAMES 2 了。■

RESCUE ON FRACTALUS

軟件名稱：RESCUE ON FRACT－ALUS
軟件類型：模擬飛行訓練
評　　分：＊＊



李德華

這遊戲的原裝版本是由雅達利所創作的，一如其他會在坊間的改編遊戲，APPLE 的版本始終是較差的。

遊戲的玩法並不是太複雜，首先玩者要選擇LEVEL （電腦本身設定爲LEVEL 4），跟着經一段飛行旅程後，便會到達地面，你要設法消滅地面的敵人，當消滅完畢後，便可升LEVEL。

遊戲提供了多個特別鍵給玩者，包括：

RESET－重新BOOT 機，但會洗去紀錄。
CTRL－R －重新執行遊戲。
ESC－暫停。
→－增加LEVEL
←－減少LEVEL
SPACE BAR－開始遊戲
→－增加速度（在遊戲中）
S －開動系統
L －降落
A－開啓機艙
B －回母艦 ■

再談 BUCKHOO BANZAI

在電腦時代上幾期中曾談過BUC－KHOO BANZAI，當時因爲沒有時間，而未能詳細地說出此遊戲的玩法，經過多月來研究後，終於把這遊戲破解，A.F.C. 的會員如想知道遊戲的破解法，可留意A.F.C. 的通訊。

閒話休說，現在應說說此遊戲的玩法，玩者在開始時應到商店拿取鏟，跟着到一塊空地裏挖掘電線，跟着再到酒店拿取電池，再者，便是到沙灘、岩石等地方拿取燃料使汽車開動，跟着便會到達一個山洞裏，那時應找尋出路，最好便是把天線連接，最後打上密碼便成功破解。以上所講的步驟全是表面化及把過程作簡單的敍述，所以讀者應留意。最值得一提的是，只有原裝的BUCK－HOO BANZAI 才能成功破解。■

APPLE FAN CLUB NEWSLETTERS

加入全港最大的蘋果電腦用家組織。閣下可以用最低廉的代價借取到無數的外國原裝軟件。
請附同一個貼上五角郵票的回郵信封及另外一個一元郵票，寄來九龍中央郵政信箱71193號電腦時代出版社收。信封面請註明「索取AFC章程及入會申請表」。

通訊欄

任天堂遊戲機大行其道

上一次在會訊中，曾介紹過高登商場最近興起MXS這部日本電腦，但高登上的「熱潮」有如「一浪接一浪」一般，MXS的地位逐漸被「FAMILY COMPUTER」所代替。FAMILY COMPUTER這牌子的電腦又是日本的產品，它之所以冒升得這麼快，主要原因是它的價格非常平，與一般的電視遊戲機的價錢差不多，但說實在的，此機種起初是專設計爲電視遊戲機的用途，而售價方面只是現在的三份二左右，但自從多個SOFTWARE公司的支持，提供一系列的SOFTWARE後，價錢便作三級跳，到現在的價錢爲¥12800。而SOFTWARE方面，我想最令大家動心而有想買此部機的衝動的，便是成龍此街機的遊戲，無疑此機種除GRAPH－IC的RESOLUTION沒有街機的高外，其他各種功能及聲音，可說是與街機不遑多讓。至於此部機的外貌，可說是非常簡單，體積只有與磁碟盒差不多大，在中間是插置CA－RTIAGE的槽，而在左右兩旁，放置了兩個隨機附送的JOYSTICK，此JOYSTICK並沒有操縱桿，有的只是與街上售賣的咭片機的按鈕，但此機是有EXTENION INTER－FACES，而JOYSTICK方面亦都有，大家可以CONN－ECT另一種專業性的JOYSTICK，例如有一隻叫HYP－ER OLYMPIC這遊戲，隨GAME是有一對專玩此GA－ME的JOYSTICK附送，來使玩此遊戲的玩者有眞實感。另外這機還設有一個有線的遙控槍，是專玩一些槍擊遊戲的，如打鴨仔、西部槍戰等。由於此機在體積方面非常細小，所以鍵盤（KEYBOARD）是另外購買連接的，而BASIC方面亦是插入一個BASIC CARTIAGE的。雖然這樣，但我想買此部機的，並不是想擁有一部電腦，而目的方面只是爲了受遊戲的吸引，希望擁有一部小型的街機而已。□

AFC收集所有熱門遊戲軟件供免費借用

遊戲軟件評論專欄每期所介紹的遊戲軟件均收藏在APPLE FAN CLU－B內的軟件庫供會員借用，軟件庫內收集有目前市面最流行熱門的軟件，加入APPLE FAN CLUB，閣下可完全免費借用所有這些軟件，包括遊戲及UTILITY查詢。參加APPLE FA－N CLUB方法，可來信九龍中央郵政信箱71193號，連同一個貼上五角郵票的回郵信封及一個一元郵票，信封面請註明「索取AFC入會表格」。

本欄資料由APPLE FAN CLUB提供。APPLE FAN CLUB（簡稱AFC是一個非牟利的蘋果電腦用家組織，每月定期出版會員通訊介紹最新軟件出版消息，遊戲軟件解拆保護的方法，BACKUP軟件參數、更改遊戲難度、機會次數等。此外每月更有大量最新軟件／使用手冊免費借給會員。索取入會章程及資料請連同回郵信封另加付一元郵票寄九龍中央郵政信箱71193號。信封面請註明「索取AFC入會章程」。□

DEPROTECT ADV. OF BUCKAROO BANZAI

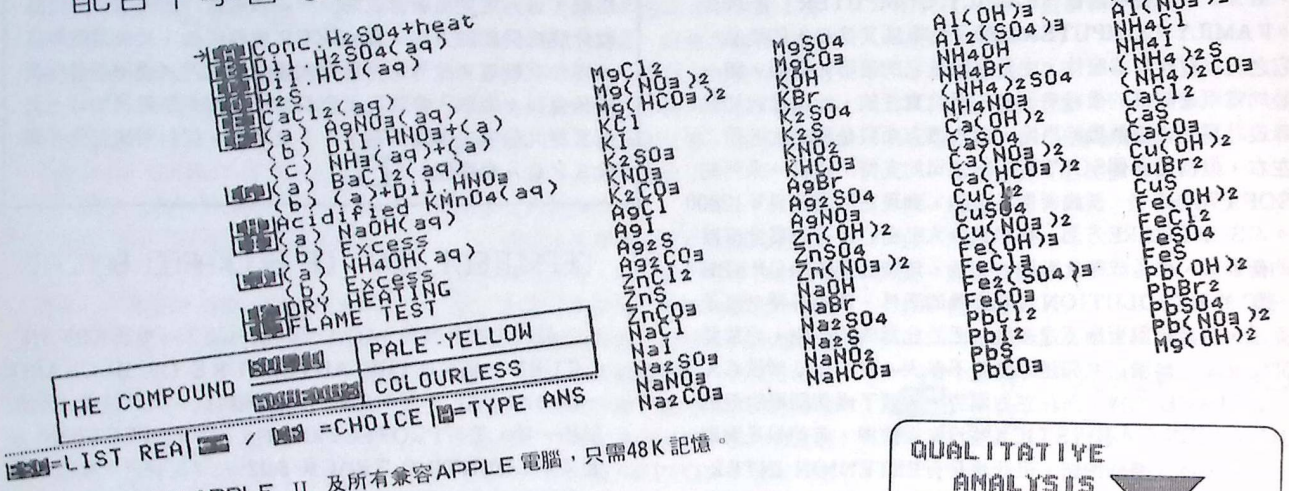
最近大家會發現SAGA又再次出品了一隻新ADVEN－TURE，名字是THE ADVENTURE OF BUCKARO－O BANZAI，而中文譯名是威龍猛探，但此遊戲又是如以往一樣，是有PROTECTION的，而PROTECTION方面又是COUNTING TRACK $22，現在我爲大家提供了一個改良爲COPYA的方法，首先是將這隻碟用LOCKSM－ITH V5.0的FASTCOPY COPY一次，或利用DISK MUNCHER亦可，然後是利用NIBBLES AWAY III或II，若是VERSION III的話，可利用SECTOR COPY，然後按7作FIND這FUNCTION，跟着是SELECT輸入狀態，即HEX或ASC II，必須要選擇HEX，然後按7作輸入ST－RING，輸入：20 00 20 然後按RETURN作搜索，但大家會發現有多個SECTOR都具有這個STRING的，我爲了使大家能自己作搜索，所以我只給一個貼士便是此SSECTOR是在TRACK $00－10處，而眞正的SECTOR可以在DISASSEMBLES作證實，在證實之前，必須將有可能的TRACK及SECTOR抄起，然後續－READ入，最後以DISASSEMBLES(即按D字)，找尋時只要找一些有程式可能的便可，所謂有程式可能的是在LIST當中COM－MAND存有BRK的存在，雖然這並不是絕對，但在此處却是非常有效，另外再給大家多一個貼士便是在JSR $2000後會跟着一個RTS，找到後只要將20 00 20改爲EA EA EA便可，而PROTECT是放在位置$2000處，而JSR $2000是如BASIC的GOSUB一樣GO SUBROU－TINE到2000此位置，然後READ TRACK $22，如果COUNT出來的BYTE與本來的一樣的話，此程式磁碟便算OK，而EA是NO OPERATION的意思，EA是它的OP CODE，而此OP CODE並不影响任何REGISTER及DATA的，亦即是說任何REGISTER如ACCUMLA－TOR內的REGISTER並不改變，這便是我們所要想得到的結果。更改完後此程式碟便不再有PROTECT了。□

中學會考無機化學化合物定性分析

模擬試驗

透過十二種標準定性分析過程，認識82種化合物特性。

配合中學會考化學科課程，適合FORM 3,4,5同學。

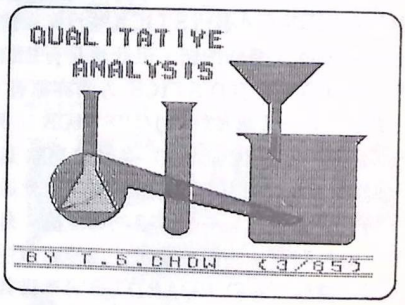


本程式適用於 APPLE II 及所有兼容APPLE電腦，只需48K記憶。

十二種標準定性分析過程

八十二種常見化合物

每套只售40元



■ 只售與長期訂戶／AFC／ ■ 電腦時代讀者服務部出售

■ 九龍旺角上海街438號同珍商業中心11字樓1104室（登打士街交界）
電話：3-312007

現年度蘋果教育基金開始接受各界申請

蘋果電腦爲提倡革新電腦教學用途而設立的「蘋果教育基金」，現已公開接受八五至八六年度申請，歡迎教師、非牟利教育機構或社團提交計劃，申請該項基金。

這項蘋果教育基金，乃由蘋果電腦香港總代理太平洋行有限公司設立，提供津貼資助，供有關機構推行有關革新電腦教學的計劃。舉凡創新教學程序、設計或發展新式的電腦應用程式，均可申請資助經費。

據太平洋行電腦教育專家兼蘋果教育基金執行幹事嘉朗素博士表示，去年獲得資助的聖約瑟書院，現已發展完成一批以圖案爲主的電腦程式，作爲校內教授核子物理的教材。此外，博愛醫院醫療隊也藉着該基金資助，購置蘋果IIe型電腦，透過電腦遊戲及圖型，輔助病人治療，訓練他們手眼配合，加速康復。

今年度的蘋果教育基金共設有六項。評選標準按申請者提交的計劃者闡述明確的理解度、計劃的價值及效率、策劃、資源和申請者資歷等區別，但最重要的是該計劃是否配合本地的教育需要，今年度尤其着重於迎合本港中學電腦教學的需要。

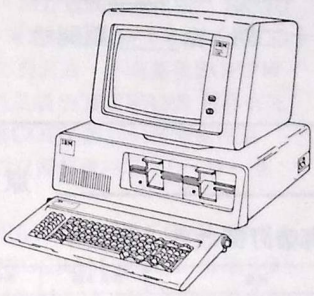
申請表格可致函香港郵政總局信箱五十六號太平洋行轉交「蘋果電腦教育基金」，或致電5-8930022 李淑娟小姐索取。■

中華基督教青年會

黃傳經中心

電腦証書課程

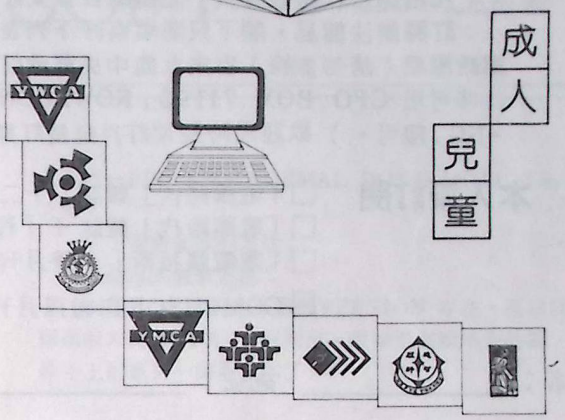
- BASIC PROGRAMMING
- COBOL PROGRAMMING
- PASCAL PROGRAMMING
- DBASE II
- MULTIPLAN
- LOTUS 1-2-3
- IBM 中文程式設計及文字處理



- 理論實習並重
- 課程內容全面

西灣河街 46—56 號西灣河大廈 2 樓
5－671571　5－671482

電腦證書課程



香 港 區：	Y.M.C.A. 青年會黃傳經中心	5-671571
	Y.M.C.A. 全青年會總會所	5-[illegible]
	香港青年協會民聯街中心	5-641277
旺 角 區：	教世軍中央堂	3-[illegible]
黃大仙區：	教世軍循道堂	3-[illegible]
紅 磡 區：	聖公會聖匠社區中心	3-[illegible]
葵 涌 區：	Y.M.C.A. 青年會所	5-[illegible]
荃 灣 區：	Y.W.C.A. 荃灣社會服務處	5-[illegible]
	電腦教育促進中心	5-[illegible]
沙 田 區：	沙角青少年中心	5-[illegible]
香港南區：	香港青年協會華富中心	5-[illegible]
屯 門 區：	Y.M.C.A. 屯門青年會	5-[illegible]
深水埗區：	軍安青少年中心	3-[illegible]
柴 灣 區：	循道衛理柴灣村服務中心	5-[illegible]

最新出版軟件

鍵入機械語言　再冇痛苦!!

假如閣下要經常鍵入機械語言列表

就不可缺少 **蘋果機械語言輸入編輯器**

設計最完美

功能最廣泛

使用最簡易

特點：

利用文字處理技術設計，使你

(1) 能隨意編輯任何的機械語言檔案
(2) 無論中間加插，改動，刪減皆易如反掌

每套只售35元（包括一份使用手冊）
本程式適合用於APPLE及APPLE相容機種
如欲購買，請親臨讀者服務部，或利用郵購（請參閱郵購需知）

Apple is the trademark of Apple Computer Inc.

省時兼便利　請即利用「電腦時代出版社」提供之訂閱服務

凡電腦時代出版社旗下之出品皆接受訂閱。

訂閱辦法簡易，閣下只需填寫好下列表格及回郵地址，連同訂閱費，利用劃線支票或郵政滙票（請勿劃線）寄來九龍中央郵政信箱 71193 號“電腦時代出版社”收卽可。（外埠可用 CPO BOX 71193, KOWLOON CENTRAL POST OFFICE, HONG KONG.卽可。）歡迎現時長期訂戶延長訂期。*如要掛號請另加付五元（每期計）。

本人擬訂閱
- ☐「電腦時代」雜誌（十二期），由第____期開始。
- ☐「電腦時代」雜誌＋「程式磁碟」（十二期），由第____期開始。
- ☐「微電腦技術」（雙月刊）十二期，由第____期開始。
- ☐ COMPUSOFT 磁碟月刊（十二期）由____期開始。

姓名：_____ 地址：_____ 電話：_____

付上劃線支票No. _____ _____ 原訂戶號碼：_____

同時請填好下列之回郵地址：

姓名：_____

地址：_____

_____

郵購訂閱收費一覽表：

種類	香港本區	澳門、中國、東南亞區	其他地區
「電腦時代」雜誌（十二期）	HK$120.00	HK$160.00（平郵） HK$280.00（空郵）	HK$190.00（平郵） HK$340.00（空郵）
「電腦時代」雜誌＋「程式磁碟」（十二期）	HK$475.00	HK$530.00（平郵） HK$690.00（空郵）	HK$570.00（平郵） HK$760.00（空郵）
「微電腦技術」（雙月刊）（十二期）	HK$110.00	HK$160.00（平郵） HK$280.00（空郵）	HK$190.00（平郵） HK$340.00（空郵）
「COMPUSOFT 磁碟月刊」（十二期）	HK$120（退碟） HK$360（連碟）	HK$396.00（平郵） HK$468.00（空郵）	HK$396.00（平郵） HK$468.00（空郵）

澳門讀者請注意：

由第廿三期開始，你們可到下列特約經銷處購買電腦時代，與及我們出版的其他叢刊。

■ 宇宙電腦電業行

雅廉訪大馬路103號地下

TEL：553689

■ 偉基利貿易發展有限公司

水坑尾2號百合大厦E座地下

（即美心餅店隔鄰或斌慧車行對面）

專爲拷貝BROTHER BUND軟件而設的POWER COPY

■ ALAN LEE ■

此程式是兩年前的作品，亦是我第一次嘗試使用COPY PROGRAM來作COPY ，而此程式大部份是由筆者朋友的帮助而寫成，所以可能有少許漏洞，但姑勿論怎樣，此程式的可靠性是佔有 80% 左右。而此程式是可供與 BROD－ERBUND SOFTWARE 公司之部份SOFTWARE的CO－PY之用，但COPY的只屬部份而論，不是整隻SOFTW－ARE 都可以COPY到，因爲這類SOFTWARE 是存有三部份PROTECTION ，而最難COPY 那部份便是使用這POWER COPY 了。而其他的只需要使用E・D・D・ Ⅲ COPY 便可。

列表(1)所LIST的PROGRAM是專用來COPY KARA－TEKA 的，如果想利用此PROGRAM COPY 其他SO－FTWARE 如DROL、GUMBALL 等，只要略加修改便行。修改方法可見文末。首先大約介紹此PROGRAM 的運行方法。

$800－862是READ的程式，讀入後首先是COMPARE ADDRESS MARK，如果是對的話，則讀入資料，反之再讀，讀完資料後再COMPARE END MARK ，如果無錯則跳出，反之再重頭讀起。
$863－8AD是 VERIFY 程式，將整隻已複制的磁碟VER－IFY 一次，如果是對的話會印出已O・K的TRACK號數，反之會停留在那TRACK處，而此部份亦是此程式的缺點，如果有朋友能修改此程式使每一條 TRACK當 VERIFY 時，不O・K的話，而會自動地將此條TRACK再COPY多一次，直至成功爲止，希望能來信賜教。

$8B0－8DC是SET UP TRACK LENGTH 及將END MARK 作AND 7F 運算，而AND 7F 便是整個PROTEC－TION 的轉捩點，亦即是PROTECTION 的心臟地帶。
$900－912是 READ ADDRESS MARK 的副程式。
$920－92F是ADDRESS MARK 的DATA。
$931－AB1是 SEARCH 程式，此段是以BIT COPY 作爲藍本，所要找尋是ADDRESS MARK的位置，並將它寫回碟內。

$B00－BFF 是文字顯示程式。
$C00－C34是主程式作控制COPY 用途，而此處是供與只有一個DRIVE 的朋友用。
$C80－CE9是主程式作控掣COPY 用途，而此處是供與只有二個DRIVE 的用家。
$D00－DB2是文字的DATA。
$E00－E25是POWER COPY 程式的起始點。

$F00－FFF 是利用NORMAL DOS 作 MOVE TRACK 之用。

以下便是修改的方法：

其他程式修改方法：

DROL 在香港是有兩個VERSION 存在，所以我將此兩個版本的DATA 一齊列出，但如果還讀不到的話，可能你手上的是另一個新版本了。

```
DROL 1
920：FD FB F6 EE EB EA DD D7 D5 BE BB
B6 B5 AF AE AD
C04 ：05
C21 ：23
C31 ：0A
C87 ：0A
CAA：05
CE1 ：23
ECE：0A

DROL 2
920：FA F6 F5 EF EE ED DE DD DA D7 BF
BB BA B6 AE AB
```

其他與DROL 1 一樣。

```
GUMBALL
920：FD F6 EF EE EB DD DB D6 D5 BE B7
B6 AE AD AB AA
C04 ：05
C21 ：23
C31 ：0B
C87 ：0B
CAA：05
CE1：23
ECE：0B
```

而以上的DATA 是放在$920處。

最後希望大家能藉此COPY PROGRAM來MODIFY 或編寫更加多的程式出來。

列表(1)打入的方法可參看「新讀者需知」。大家於打入完列表(1)後可用下列指令存入磁碟。

```
BSAVE POWER COPY，A$7FD，L$803
```

或者在同期出版的程式磁碟中找到相同名字的檔案，BRUN 亦可。 ■

專爲拷貝 BROTHERBUND 軟件而設的 POWER COPY

```
::::::::::::::::
: POWER COPY   :
: A67FD,L$803  :
:      BY      :
:   ALAN LEE   :
: COPYRIGHT(C) :
:   1985 BY    :
:  COMPUHOUSE  :
::::::::::::::::
07FD- 4C 00 0E
0800- A2 0C 86 54 A0 00 8C 48
0808- 08 84 55 A2 60 EA BD 8C
0810- C0 10 FB C5 50 D0 F7 BD
0818- 8C C0 10 FB C5 51 D0 EE
0820- BD 8C C0 10 FB C5 52 D0
0828- E5 A4 55 B9 00 BB 8D 55
0830- BB E6 55 BC 8C C0 10 FB
0838- B9 00 BC 0A 0A 0A 0A BC
0840- 8C C0 10 FB 19 00 BC 8D
0848- 00 3C EE 48 08 B0 E4 EE
0850- 49 08 BD 8C C0 10 FB C5
0858- 53 D0 A2 C6 54 D0 CA 2C
0860- E8 C0 60 A9 30 8D 49 08
0868- E6 0A 4C 00 08 2C 32 32
0870- A0 00 8C 54 BB 84 55 A2
0878- 60 EA BD 8C C0 10 FB C5
0880- 50 D0 F7 BD 8C C0 10 FB
0888- C5 51 D0 EE BD 8C C0 10
0890- FB C5 52 D0 E5 BD 8C C0
0898- 10 FB 91 00 E6 00 D0 F5
08A0- E6 01 A5 01 C9 50 90 ED
08A8- BD 88 C0 4C 00 0A 4B 3D
08B0- A9 00 85 00 8D A0 41 85
08B8- 05 A9 29 85 01 85 06 A2
08C0- 0C A0 00 E6 06 E6 06 B1
08C8- 05 C5 53 F0 08 C8 C0 20
08D0- D0 F5 4C F0 08 29 7F 91
08D8- 05 CA D0 E5 60 4C 4F 43
08E0- 4B 3D 32 20 55 4E 4C 4F
08E8- 43 4B 3D 33 20 44 45 4C
08F0- A5 04 4A 20 DA FD 20 48
08F8- F9 20 2D FF 4C 8E FD D0
0900- A5 04 4A A2 03 29 0F A8
0908- B9 20 09 95 50 C8 98 CA
0910- 10 F3 60 00 00 00 00 00
0918- 00 00 00 00 00 00 00 00
0920- FD FB F7 F6 F5 EE EB EA
0928- DE DD D7 D6 BB AF AE AD
0930- 03 A9 00 8D 4B 03 20 BB
0938- 0A AD 7D 03 85 04 20 5F
0940- 1C A9 D7 8D 54 03 20 95
0948- 1B 20 6F 10 AD 7D 03 29
0950- F0 AA BD 8D C0 BD 8E C0
0958- 10 0A A4 08 A9 04 99 25
0960- 09 4C 97 0A A4 17 BD 8D
0968- C0 BD 8E C0 A9 BE 9D 8F
0970- C0 DB 8C C0 20 D6 09 B1
0978- 00 09 80 EA EA 9D 8D C0
0980- DB 8C C0 C4 00 C8 D0 21
0988- E6 01 B1 00 EA F0 41 09
0990- 80 9D 8D C0 DB 8C C0 4C
0998- 85 09 09 80 9D 8D C0 DB
09A0- 8C C0 EA EA 4C 85 09 EA
09A8- EA B1 00 F0 23 10 05 C4
09B0- 00 4C 7D 09 09 80 24 00
09B8- 9D 8D C0 DB 8C C0 EA EA
09C0- 4C 85 09 EA EA 24 00 9D
09C8- 8D C0 DD 8C C0 4C 85 09
09D0- DD 8E C0 DD 8C C0 60 00
09D8- 8D 4C 03 DD 88 C0 AD 00
09E0- C0 C9 9B D0 10 AD 10 C0
09E8- 68 68 A9 00 8D 57 03 20
09F0- 1E 28 4C 79 26 A4 08 B9
09F8- 25 09 C9 02 F0 05 AD 31
0A00- A9 00 85 05 A9 28 85 06
0A08- A0 00 B1 05 C5 50 D0 70
0A10- 20 A0 0A B1 05 C5 51 D0
0A18- 67 20 A0 0A B1 05 C5 52
0A20- D0 5E 84 05 38 A5 05 E9
0A28- 50 85 00 A5 06 E9 00 85
0A30- 01 18 A5 00 69 A0 85 02
0A38- A5 01 69 18 85 03 A0 00
0A40- A9 00 91 02 A5 01 85 FF
0A48- A9 00 85 00 4C BF 08 36
0A50- C1 33 39 3A B0 31 30 30
0A58- 30 3A 42 24 D0 42 24 C8
0A60- E7 28 43 29 3A 82 43 48
0A68- 3A 96 31 3A 8C C9 38 36
0A70- 38 3A BA 42 24 3A BA 44
0A78- 24 3B 42 24 3A AB 31 30
0A80- 20 A0 0A 4C 0A 0A 43 31
0A88- D0 B3 28 43 56 CB 38 29
0A90- 3A 43 32 D0 43 56 C9 43
0A98- 31 CA 38 3A 50 B0 31 30
0AA0- C8 D0 1D E6 06 A5 06 C9
0AA8- 43 D0 15 2C E9 C0 20 40
0AB0- 03 4C 00 0A B0 E2 28 50
0AB8- 29 3A B1 00 00 00 0A 44
0AC0- 60 01 FF 95 00 00 42 00
0AC8- 85 28 00 00 00 54 00 85
0AD0- 38 00 00 00 43 48 83 00
0AD8- 00 00 00 43 56 85 40 00
0AE0- 00 00 43 31 82 00 00 00
0AE8- 00 43 32 83 60 00 00 00
0AF0- 50 00 8B 7A 80 00 00 43
0AF8- 00 88 20 00 00 00 53 00
0B00- 20 58 FC A0 88 B9 00 0D
0B08- EA EA 20 F0 FD C8 C0 B0
0B10- D0 F3 A9 8D 20 F0 FD EA
0B18- A9 8D 20 F0 FD 2C 10 C0
0B20- A0 19 B9 00 0D 20 F0 FD
0B28- C8 C0 2D D0 F5 20 8E FD
0B30- AD 00 C0 10 FB C9 B1 F0
0B38- 2A C9 B2 D0 DA 4C 80 0C
0B40- 2C 10 C0 20 0C FD 4C 8E
0B48- FD A9 0F 20 66 0B 00 00
0B50- 20 8E FD A0 2D B9 00 0D
0B58- 20 F0 FD C8 C0 3F D0 F5
0B60- 4C 40 0B 4C 00 0C 20 5B
0B68- FB 4C 8E FD A9 05 85 07
0B70- 20 50 0E A9 0C 20 5B FB
0B78- A0 3F B9 00 0D 20 F0 FD
0B80- C8 C0 4D D0 F5 A5 07 4C
0B88- C8 0B 20 40 03 A9 0F 20
0B90- 5B FB A0 4D B9 00 0D 20
0B98- F0 FD C8 C0 58 D0 F5 20
0BA0- D0 0B A9 0F 20 5B FB A0
0BA8- 00 A9 A0 20 F0 FD C8 C0
0BB0- 20 D0 F6 4C 8E FD A9 0C
0BB8- 20 5B FB A0 58 B9 00 0D
0BC0- 20 F0 FD C8 C0 66 D0 F5
0BC8- A5 07 20 DA FD 4C 8E FD
0BD0- 2C 10 C0 20 0C FD 4C 8E
0BD8- FD A9 0C 20 5B FB A0 69
0BE0- B9 00 0D 20 F0 FD C8 C0
0BE8- 7A D0 F5 4C 8E FD EA EA
0BF0- 4C 8E FD 2C 10 C0 20 0C
0BF8- FD 4C 8E FD FF FF 00 00
0C00- 20 2D 0C A9 04 85 07 2C
0C08- E9 C0 20 73 0B 20 40 03
0C10- 20 8D 0B 2C E9 C0 20 B6
0C18- 0B 20 27 0C E6 07 A5 07
0C20- C9 21 D0 E1 4C 80 0E 20
0C28- 90 03 4C 8D 0B 20 50 0B
0C30- A9 08 85 04 4C 00 03 00
0C38- FF FF 00 00 FF FF 00 00
0C40- FF FF 00 00 FF FF 00 00
0C48- FF FF 00 00 FF FF 00 00
0C50- FF FF 00 00 FF FF 00 00
0C58- FF FF 00 00 FF FF 00 00
0C60- FF FF 00 00 FF FF 00 00
0C68- FF FF 00 00 FF FF 00 00
0C70- FF FF 00 00 FF FF 00 00
0C78- FF FF 00 00 FF FF 00 00
0C80- EA EA EA 20 50 0B A9 08
0C88- 85 04 2C EA C0 20 00 03
0C90- AD 78 04 85 08 2C E8 C0
0C98- 20 A8 FC 2C EB C0 20 00
0CA0- 03 AD 78 04 85 09 2C E8
0CA8- C0 A9 04 85 07 2C E9 C0
0CB0- A5 08 8D 78 04 20 30 0E
0CB8- EA EA EA 20 40 03 AD 78
0CC0- 04 85 08 2C E9 C0 A5 09
0CC8- 8D 78 04 20 A8 FC 20 40
0CD0- 0E 20 F0 0C 20 90 03 AD
0CD8- 78 04 85 09 E6 07 A5 07
0CE0- C9 21 D0 C7 20 D9 0B 4C
0CE8- 80 0E 00 00 FF FF 00 00
0CF0- A0 05 A9 FF 20 A8 FC 88
0CF8- 10 F8 60 00 FF FF 00 00
0D00- A0 A0 A0 A0 A0 A0 A0 A0
0D08- 10 0F 17 05 12 20 03 0F
0D10- 10 19 20 AB 60 43 4F 50
0D18- 59 B1 A0 CF D2 A0 B2 A0
0D20- C4 C9 D3 CB A0 C4 D2 C9
0D28- D6 C5 D3 A0 BF C9 CE D0
0D30- D5 D4 A0 CF D2 C7 C9 CE
0D38- C1 CC A0 C4 C9 D3 CB 52
0D40- 45 41 44 49 4E 47 A0 D4
0D48- D2 C1 C3 CB A0 C3 C8 C1
0D50- CE C7 C5 A0 C4 C9 D3 CB
0D58- 57 52 49 54 49 4E 47 A0
0D60- D4 D2 C1 C3 CB A0 DC A0
0D68- A0 56 45 52 49 46 59 A0
0D70- D4 D2 C1 C3 CB A0 A0 A0
0D78- A0 A0 A0 A0 A0 A0 A0 A0
0D80- A0 A0 A0 A0 A0 A0 A0 A0
0D88- 10 0F 17 05 12 20 03 0F
0D90- 10 19 20 AD AD BE A0 C2
0D98- D9 A0 57 4F 4E 47 60 48
0DA0- 4F 60 4B 41 4D A0 A6 A0
0DA8- C1 CC C1 CE A0 CC C5 C5
0DB0- A0 A0 00 00 FF FF 00 00
0DB8- FF FF 00 00 FF FF 00 00
0DC0- FF FF 00 00 FF FF 00 00
0DC8- FF FF 00 00 FF FF 00 00
0DD0- FF FF 00 00 FF FF 00 00
0DD8- FF FF 00 00 FF FF 00 00
0DE0- FF FF 00 00 FF FF 00 00
0DE8- FF FF 00 00 FF FF 00 00
0DF0- FF FF 00 00 FF FF 00 00
0DF8- FF FF 00 00 FF FF 00 00
0E00- A2 00 BD 00 0F 9D 00 03
0E08- E8 D0 F7 A9 63 8D FE 07
0E10- A9 08 8D FF 07 4C 00 0B
0E18- 85 3F A9 FE 85 3E A0 00
0E20- 20 2C FE 4C 00 0B 00 00
0E28- FF FF 00 00 FF FF 00 00
0E30- 2C EA C0 20 73 0B 60 00
0E38- FF FF 00 00 FF FF 00 00
0E40- 2C EB C0 20 B6 0B 60 00
0E48- FF FF 00 00 FF FF 00 00
0E50- A9 0A 85 04 60 FF 00 00
0E58- FF FF 00 00 FF FF 00 00
0E60- FF FF 00 00 FF FF 00 00
0E68- FF FF 00 00 FF FF 00 00
0E70- FF FF 00 00 FF FF 00 00
0E78- FF FF 00 00 FF FF 00 00
0E80- 20 CA 0E EA 20 AC 0E 20
0E88- A8 FC 20 A8 FC A5 04 4A
0E90- 20 BF 0E 20 40 03 20 FD
0E98- 07 A5 0A C9 43 41 00 00
0EA0- 00 00 00 A5 04 C9 42 D0
0EA8- DB 4C D2 0E A9 00 85 0A
0EB0- 2C E9 C0 60 20 2D FF A9
0EB8- A0 20 ED FD 4C 9F 0E A9
0EC0- A0 20 ED FD A5 04 4A 4C
0EC8- DA FD 20 00 03 A9 08 85
0ED0- 04 60 2C 10 C0 20 48 F9
0ED8- 20 3A FF 20 0C FD 4C 00
0EE0- 0B FF 00 00 FF FF 00 00
0EE8- FF FF 00 00 FF FF 00 00
0EF0- FF FF 00 00 FF FF 00 00
0EF8- FF FF 00 00 FF FF 00 00
0F00- EA EA EA EA EA AD E9 C0
0F08- 4C 13 03 A0 03 B9 E9 C0
0F10- AD E9 C0 A2 60 A9 50 20
0F18- 95 BE A9 28 85 45 A9 00
0F20- 4C 5A BE A5 FF 85 01 A2
0F28- 60 A0 00 20 66 09 E6 04
0F30- E6 04 2C E8 C0 60 4C 40
0F38- 03 00 00 00 00 00 00 00
0F40- A9 00 85 00 A9 10 85 01
0F48- A5 04 A2 60 20 B6 03 20
0F50- 00 09 4C 70 08 00 E6 00
0F58- D0 F5 E6 01 A5 01 C9 60
0F60- 90 ED AD E8 C0 60 AD E9
0F68- C0 AD EE C0 60 01 3E 4C
0F70- 02 03 20 04 05 06 FF D0
0F78- CD A9 00 8D 07 08 20 E3
0F80- 01 60 01 00 11 00 91 03
0F88- 00 95 00 00 01 00 FE 60
0F90- 2C E9 C0 20 A8 FC EA EA
0F98- A5 04 A2 60 20 C8 03 A0
0FA0- 00 20 66 09 A0 00 A2 60
0FA8- 20 A8 FC 4C 23 03 C9 30
0FB0- 90 EF 20 66 09 60 A0 00
0FB8- 20 A0 B9 A0 00 60 BD 8E
0FC0- C0 BD 8C C0 BD 88 C0 60
0FC8- 20 A0 B9 A9 FF EA EA EA
0FD0- EA EA EA 48 68 EA 60 EF
0FD8- C0 60 B5 B7 AD 0F 9D AC
0FE0- 0E 9D 60 AD C2 AA AC C1
0FE8- AA 60 4C 51 A8 EA EA 4C
0FF0- 2C E9 C0 4C 40 03 58 FF
0FF8- 4C 65 FF 4C 65 FF 65 FF
1000- C8 31
```

2732上的SUPER ROM

■ 就

電腦時代第廿三期刊登過一個SUPER ROM 廣告，此ROM可代替原本APPLE（第三代版）的F8，而擁有類似WILDCARD 及其他多種功能。經常研究拆保護的發燒友必定覺得非常實用。而且使用此枚SUPER ROM 可以省回一個SLOT，同時不會因爲某些要用64K 記憶的程式軟件難倒。因爲使用WILDCARD 的同時，必須要有16K RAM咭。不過，此枚SUPER ROM 的唯一缺點，就是當運行某些如PRODOS 等要CHECK ROM 的軟件時，就必須原本的F8了。雖然電腦時代廿三期同時刊出一篇制作DUAL ROM CARD，可供用家選擇，不過筆者發覺另一個更簡單的方法，而且較爲慳錢的製作，就是用一枚2732來代替兩枚2716。根據資料手冊顯示，2732與2716是可相容及同是24脚的，在存放資料方面2716爲2K而2732是4K，相等於兩枚的2716。

在這枚2732上頭2K存放原本F8內容（F800－FFFF），而尾2K是存放SUPER ROM的資料（亦是 F800－FFFF）。最後由2732中帶出選擇控制，便可隨便使用以上兩種F8 功能而無須拔插和翻拆機件，亦不需要DUAL ROM CARD。

要製造一枚雙重功能的2732，必須要有下列工具：

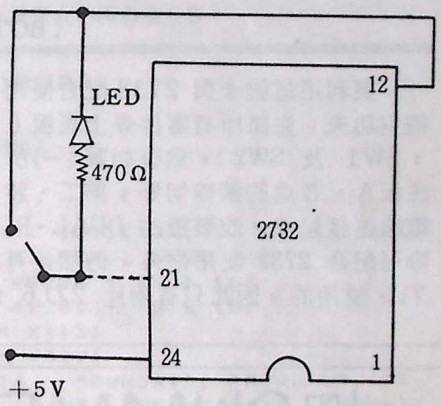
(1) 一塊EPROM WRITER
(2) 一枚2732
(3) 一個單刀雙擲選擇掣
(4) SUPER ROM資料

修改的工作程序：

(1) 將SUPER ROM資料存入記憶中
（e、g BLOAD SUPER ROM，A$2000）
(2) 將原本F8資料存入記憶中
（e、g 1800<F800・FFFFM）

編者註：SUPER ROM只適用於第三代版蘋果電腦，不適合用於第五代雙CPU機種，不過讀者服務部另備有適合第五代版機種使用的SUPER ROM Ⅱ，請來電查詢詳情。

(3) 將所有資料由記憶中寫入EPROM WRITER 內的2732裏面。



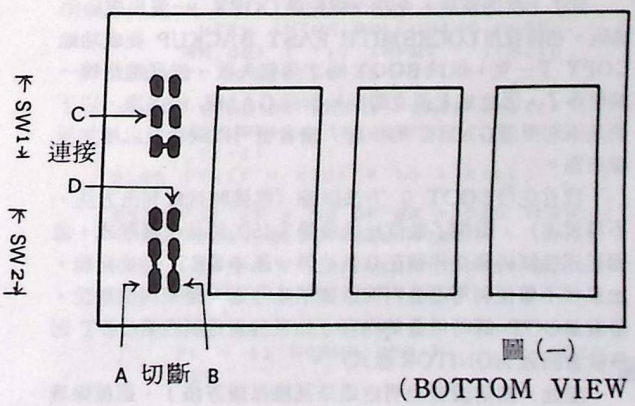
（e、g WRITE：ADDRESS FROM 1800）註：2732可存放4K 資料由1800至27FF。頭2K爲原本F8（1800－1FFF）尾2K爲SUPER ROM內容（2000－27FF）

(4) 由2732中，領出選擇線及焊好選擇掣。在2732上的12脚和24脚是選擇線，而21脚是中線，同時不用插入ROM架內。以下爲該線路圖：（註：21脚不可插入ROM SLOT 內）

■

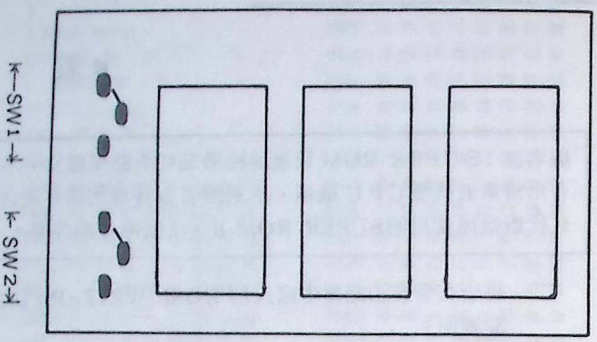
記得上一期「電腦時代」提到製造 DUAL-ROM-CARD 的時候，筆者設計的底板綫路是有兩個雙刀雙擲式的開關。當時筆者故作神秘，現在可以將其中巧妙處告訴讀者了。

大家若果留心的話，就會發覺今期「電腦時代」有一篇文章是介紹如何將 SUPER-ROM 及 F8-ROM 的內容同時存放在一片 2732 上。有了這種方法，我們就可以在一片晶片上同時擁有 SUPER-ROM 及 F8-ROM。選擇的方法亦只是利用一個簡單的開關。筆者有先見之明，早在設計這張卡時就照顧到只擁有這片 2732 的讀者了。



圖（一）
（BOTTOM VIEW）

SUPER ROM



圖(二)
(BOTTOM VIEW)

要利用這張卡與 2732 配合使用，讀者需要做些少功夫。先將所有零件焊上底板（包括兩個開關：SW1 及 SW2），然後如圖(一)所示，第一、將底板A、B處的銅線切斷；第二、將C、D兩點用電線連接起來。改裝後的 DUAL-ROM-CARD，除可配合 2732 使用之外，仍然是可以配合兩片 2716 使用的。因此只有兩片 2716 的讀者亦可以依法改裝，得日後取得上述 2732 後便可輕易裝上。要注意的是：在配合 2732 使用時，2732 只可插在 ROM 1 的 IC 座上；而 ROM 2 則讓它空着或仍可以插一片 2716，視乎需要而定。

要知道改裝後 DUAL-ROM-CARD 的開關選擇位置，可參看圖(二)及表(一)。讀者若擁有兩片 2716 分別插在 ROM 1 及 ROM 2的位置，只需將 SW2 推至（IV），而此時 SW1 分別用作選擇 ROM 1 及 ROM 2 。如果是 2732（必須插在 ROM 1 處），則需將 SW2 推至（III）的位置，由 SW1 來選擇 2732 的兩部份。最後一點要提醒各位讀者，選擇開關前請先將電腦關掉，避免對 ROM 造成損壞。■

表(一)

SW1 / SW2	I	II
III	2732 的上半部	2732 的下半部
IV	ROM 2 (2716)	ROM 1 (2716)

把SUMMER GAME II 改成COPYA 版本

■ 周志成

SUMMER GAME Ⅱ一推出已極受歡迎，因爲上集有很好的表現，當筆者玩這個GAME時也十分滿意，唯獨是獨木舟一項目比較難控制，其他項目都是水準之作。

由於這遊戲圖畫十分吸引，筆者的弟妹也嚷着要玩，更趁我不在時自己取碟BOOT來玩，而把我這隻SUMMER GAME Ⅱ弄DOWN了，累得我要再COPY 回家，白白花多了十元（包括車資）。

有了上次的經驗，今次一開始便COPY 一隻出來給弟妹玩，初時我用LOCKSMITH FAST BACKUP 便成功地COPY 了一隻，但試BOOT時才發覺失敗，磁碟機空轉一刻便停了，因此筆者便立即開始拆這GAME 的保護，以下便是筆者拆這GAME 的步驟，讀者也可以跟着做以增加拆碟造詣。

我首先以BOOT 0 方法跟踪（電腦時代曾刊出方法，不再重述），跟踪了數位址後發覺$1500便是保護程式，這個程式控制磁碟機停留在0軌找尋一連串$E7 的位元組，如果找不着便利用堆疊內的數據胡亂回返。後果則很難說，要看BOOT 碟時堆疊的內容（如果讀者是關機BOOT 則多數會回返MONITOR程式）。

至此，相信讀者也明白這是甚麼保護方法了，這種保護方法是於磁區與磁區中加入一些特定值的位元組，再以程式找尋這些位元組是否存在，若沒有則不再讀取資料。

那爲甚麼用FAST BACKUP COPY 會失敗，因爲FAST BACKUP 這一類COPY程式祇會COPY磁區內資料而忽略了磁區與磁區中的資料，因而溜了這一批特定的位元組，如果想COPY 這隻碟十分容易，祇要利用BIT COPY 那一類程式補回零軌便行（高登便是以此方法COPY）。

用上述的方法COPY雖然成功，但十分麻煩而且不十分準確，所以最好能改成FAST BACKUP 直抄更理想。

我們可使用TRACK EDIT那一類工具觀看零軌內容，發覺果然有一連串$E7存在，另外一連串$FF 爲同步位元組及一連串$96爲空磁區，而這GAME 的保護程式是找一連串$E7，我們可把它改成找尋$FF或$96便可拆解保護成爲COPYA版本。

讀者先用FAST BACKUP COPY一隻，成功後再用SECTOR EDITOR把TRACK 0 ，SECTOR 1 ，第$4A，$53，$5C 三個BYTE的內含由$E7改爲$96，再WRITE回磁碟上便成功。這隻磁碟以後便可以FAST BA－CKUP 直抄而不須補TRACK，最後祝讀者得心應手。■

發揮 CLOCK 咭的潛力

加入時鐘咭功能的私人約會備忘日曆

K.K

電腦時代第五期所刊出的私人約會日記部（P、A、C、），在使用上相當方便，在下對其唯一不滿之處，是在每天使用前都要輸入當天日子。筆者擁有一張CLOCK CARD 或稱TIME II CARD多時，起初利用了卡上的日期，減少了每天輸入日子的麻煩，使用了一段日子後，又覺得不充份利用鐘卡，多少有點浪費，於是動手把鐘卡所記錄的年、月、日、星期、時間，全加在P、A、C、上。

該介面卡是以數字形式去記錄日子的，所以要由程式來把數值化成文字，如星期幾、AM或PM等，這些工作由程式行10－30及10000－10340來完成，若果鐘卡不是插在SLOT 5，請改變程式行12之S值。若果閣下觀看呼叫鐘卡的副程式，會發現有多個 POKE B,0 的指令，B＝49281＋（S＊16），S是SLOT 的值，該指令是程式把主控權交回鐘卡，由鐘卡內部的計時器繼續計時，若果不把主控權交回鐘卡，那麼，祝閣下好運！用鐘卡附送的TIME II SET 程式，把日期、時間再輸入吧！亦因爲這個原因，當閣下要離開P、A、C、時，請使用第五項，由程式來把主控權交回鐘卡，不要直接關掉主機。

若要加上保護，請更改下列兩句：

```
9000 TEXT:HOME:NEW
9315 IF ER=56 OR ER=255
     THEN POKE B,0:CALL
     43089:CALL 1400
```

現在按下 RESET 或 CTRL－C 都會 BOOT 機，CALL－1400 會使磁碟機從新起動，載入DOS ，若果不想 BOOT 機，可把 CALL－1400 改爲NEW。

P、A、C、原程式有另一個缺點，當使用第二、三項時，程式會問及日期，輸入1,1,85和輸入1, 1, 95都沒有分別，原因是程式不理會年份，所以筆者把輸入日期簡化，只需輸入月、日即可，年份是一年內的，所以可由鐘卡取出。

鐘卡的說明書十分簡單，在下對其使用不很清楚，希望筆者的些少修改能提起其他人士對該介面的興趣，發展更多實用程式，提供更多該介面的使用說明，使其物有所值，多謝！

程式輸入

由於本文原本程式已在電腦時代第五期第13頁刊登過，爲了節省篇幅，不擬在此將原文程式重登一次，我們只列出要增刪的行號部份供大家參考更改。

不過大家可在同期出版的程式磁碟中找到一個經修改過的完整程式，檔案名字爲P. A. C. CLOCK VERSION。

編者按：多謝K K君的來稿，將 CLOCK CARD 應用在我們的程式上。電腦時代本著同讀者分享研究蘋果電腦的宗旨，經常呼籲大家投稿，特別是一些個人的研究心得。CLOCK CARD在各種咭仔中是較爲少人提及的，事實上據我們所知，這張咭應有很大的發展用途潛力。我們希望曾對CLOCK CARD應用下過一番研究功夫的讀者能將經驗寄來給我們，好讓廣大的讀者能分享。

原程式刪去的行號包括：
370～400
增加的新行號包括：
10～30及85

```
10  REM   CLOCK
12 S = 5: REM   SLOT = 5
14 A = 49282 + (S * 16)
16 B = 49281 + (S * 16)
18  DIM X(13)
20  DIM Y$(7)
22  DATA   "SUNDAY", "MONDAY", "T
    UESDAY", "WEDNESDAY", "THURS
    DAY", "FRIDAY", "SATURDAY"
24  FOR C = 0 TO 6
26  READ Y$(C)
28  NEXT C
30 I = 0

85  POKE 1010,233: POKE 1011,242:
    CALL  - 1169
```

需要更改的行號：

```
410  VTAB 21: HTAB 5: PRINT "SELE
     CT ONE OF THE ABOVE:_": GOTO
     10000

440  TEXT : HOME :FL = 1: GOSUB 2
     50: GOTO 320

3010  HOME : VTAB 12: PRINT "ENTE
     R DATE FOR MEMOS YOU WISH TO
      VIEW INTHE FOLLOWING FORM (
     MM,DD) : ";: INPUT "";TM,TD
3015 TY = X(0) * 10 + X(1)

5160  VTAB 7: PRINT "DATE #";I;"
     (MM,DD) : ";: INPUT "";TM(I)
     ,TD(I)
5165 TY(I) = X(0) * 10 + X(1)

9315  IF ER = 56 OR ER = 255 THEN
      POKE B,0: CALL 43089

9350  IF ER = 22 THEN FL = 1: GOSUB
     250:FL = 0: GOTO 320

9400  IF ER = 254 AND EL > 300 THEN
     FL = 1: GOSUB 250:FL = 0: GOTO
     320
```

加八時鐘卡功能的P·A·C·

```
10000  REM   CLOCK
10010 I = 0
10020  POKE B,16
10030  FOR E = 44 TO 32 STEP  - 1

10040  POKE A,E
10050 X(I) =  PEEK (A)
10060 I = I + 1
10070  NEXT E
10080  IF F > 7 THEN P$ = " "
10090  IF X(4) > 3 THEN X(4) = X(
     4) - 4
10100 F = X(7)
10110  IF F > 7 THEN P$ = " "
10120  IF X(7) > 7 THEN X(7) = X(
     7) - 8
10130  IF F < 8 AND X(7) < 4 THEN
     P$ = " A.M."
10140  IF F < 8 AND X(7) > 3 THEN
     P$ = " P.M."
10150  IF X(7) > 3 THEN X(7) = X(
     7) - 4
10160  VTAB 10
10170  PRINT X(2);X(3);"/";X(4);X
     (5);"/";X(0);X(1);
10180  HTAB (14)
10190  PRINT Y$(X(6));
10200  HTAB (28)
10210  PRINT X(7);X(8);":";X(9);X
     (10);":";X(11);X(12);P$
10220  IF  PEEK ( - 16384) > 127 THEN
      GOTO 10260
10230  POKE  - 16368,0
10240  POKE B,0
10250  GOTO 10010
10260  POKE B,0
10270 AA = ( PEEK ( - 16384) - 17
     6): POKE  - 16368,0
10280  IF AA < 1 OR AA > 5 THEN 1
     0010
10290 A$ =  STR$ (AA)
10300 MM = X(2) * 10 + X(3)
10310 DD = X(4) * 10 + X(5)
10320 YY = X(0) * 10 + X(1)
10330 YR$ =  STR$ (YY)
10340  GOTO 420
```

■

FORMATTING DISK 雜談

呂允健

本人在貴刊看到近幾期刊出了不少有關四十磁軌應用的程式及文章，本人起初不知如何製造四十軌的磁片，幾經辛苦才發現如何格式化（INIT）40軌的磁片。相信不少讀者也會有同樣的困難，因此本人特在此談一談。

磁片格式化利用了電腦本身的兩個副程式：DSKFORM（$BEAF－$BF0C）和RWTS 中的DISKF2（$BF0D－$BFA7）。在DSKFORM程式中位址$BEBC 記錄了格式化最終的磁軌，通常這位址上的值是$23（35），我們只需將$BEFE 的值改爲$28便可FORMAT出40軌的磁片。

相信不少讀者都不明白程式磁片如何能受到保護，其實一般人亦可製造出防盜磁片（不能被拷貝的磁片），原理是利用非標準的格式化、同步BYTE或更改RWTS 均可。筆者今次準備寫製非標準格式化磁片的方法。

```
〕CALL－151
*BCAE：C9
*B991 ：C9
*BCB3：BB
*B99B ：BB
*CTRL－C RETURN
〕INIT HELLO
```

讀者只需先LOAD入－HELLO程式，再照上述方法鍵入，便可製造出一隻防盜磁片，這張磁片對一般拷貝程式也有少許防止拷貝作用，當然效果未必十分好，也有參考的價值。■

加强版本的鍵盤發聲程式

黃博思

這程式只有36個BYTE ，除了使閣下的鍵盤像IBM的一樣能在按鍵時發出嗶嗶聲外，還把原來APPLE 的反白游標改爲"－"。最好的是它無論在APPLESOFT BASIC，INTEGER BASIC 和MONITOR 上都能操作，兼且程式極度的短，可以移在DOS內連BOOT 機時一起讀入，移入DOS的方法可見電腦時代。新的發聲游標用FP，NEW等都不能變回原來面目，但可以利用BOOT 機（PR＃6）和RESET，游標便回復正常。

最後，須注意的是若閣下使用的鍵盤是擁有功能鍵的話（即是說按CTRL－6會出PR＃6，CTRL－8出CATA L－OG 等功能）本程式便會使原有的功能鍵失效，功能鍵用者請注意。

程式只要照下面列表鍵入，再用BSAVE UNDER LI－NE BEEP CUSOR，A$300，L$36 便可錄入碟，要用時只要BRUN 便成。讀者亦可在同期出版的程式磁碟中找到同名字的一個FILE。■

```
: UNDERLINE BEEP CUSOR :

0300- A9 0B 85 38 A9 03 85 39
0308- 4C EA 03 91 28 B1 28 48
0310- A9 DF 91 28 68 91 28 2C
0318- 00 C0 10 F1 AD 00 C0 2C
0320- 10 C0 48 8A 48 A2 20 A9
0328- 08 20 A8 FC AD 30 C0 CA
0330- D0 F5 68 AA 68 60 00 00
```

■

有預報功能的新 INIT 指令

在 FORMAT 磁碟前會有所警告

■張峯寧

各位讀者，在操作電腦時，若有錯誤發生，可能會導致很大的損失！尤其在使用ＩＮＩＴ這指令時，概因要知道ＩＮＩＴ這指令是會將一隻磁碟（不論是否存有資料）重新規劃ＦＯＲＭＡＴ，換句話說，你可能會將一隻存有非常重要及有價值資料的磁碟由於使用上的錯誤ＩＮＩＴ了，而導致所有ＤＡＴＡ的失去！試想，當你遇到上述情況時，你會怎樣？事實上，發生上述情況的機會是很大的，所以我們不得不想一法子來補救，但到現時爲止，筆者仍未想出恢復磁碟上被ＦＯＲＭＡＴ後而失去之資料（筆者相信這是沒有可能的），不過筆者却想出一個預防的方法；所謂預防勝於治療，況且預防也是唯一可行之途徑！

方法很簡單，只須利用列表一及二之程式即可擁有ＦＡＩＬＳＡＦＥ ＩＮＩＴ 功能了！ＦＡＩＬＳＡＦＥ ＩＮＩＴ 主要是將舊ＩＮＩＴ功能改良，若要ＦＯＲＭＡＴ的磁碟是有資料的話，程式即會提示用家，使用家有挽救有重要資料之磁碟的機會！

但是，針無兩頭利，由於ＩＮＩＴ指令的加強，另一方面，你必須放棄一些東西！由於有了ＦＡＩＬＳＡＦＥ ＩＮＩＴ，ＦＰ及ＩＮＴ指令便被取消了！而且程式必須在４８Ｋ或更大以上的記憶體的蘋果電腦上運行；並且只能ＤＯＳ３．３系統下運行！

使用ＦＡＩＬＳＡＦＥ ＩＮＩＴ：

要使新ＩＮＩＴ指令，只須鍵入列表一及二，再ＲＵＮ ＩＮＩＴ ＰＯＫＥＲ；在ＩＮＩＴ ＰＯＫＥＲ 程式中會首先廢除ＦＰ及ＩＮＴ之功能。

好了！現在ＲＵＮ ＩＮＩＴ ＰＯＫＥＲ，再放入一隻ＩＮＩＴ過的磁碟，然後鍵入ＩＮＩＴ ＨＥＬＬＯ，此時由於程式查到這碟是有資料的，於是便會將之ＣＡＴＡＬＯＧ出來，此時若你發現此碟是有用的，便可以趁機換之！之後再按一下，程式便會執行ＩＮＩＴ的工作——ＦＯＲＭＡＴ磁碟。換句話說，此新ＩＮＩＴ指令可以被稱爲一個有提示的ＩＮＩＴ！另一方面，若你是放入一隻未經ＩＮＩＴ的磁碟，則程式是不會有任何提示而將之ＩＮＩＴ。

請注意，此程式只會「保護」用ＤＯＳ３．３規劃的磁碟，若你放入ＣＰ／Ｍ、ＰＲＯＤＯＳ、ＰＡＳＣＡＬ 之規劃磁碟，程式會將之當爲不是用ＤＯＳ３．３規劃，而將之ＩＮＩＴ！

至於鍵入列表一之方法可參閱１７期「機械碼輸入方法」，鍵入後可以ＢＳＡＶＥ ＦＡＩＬＳＡＦＥ．ＩＮＩＴ，Ａ＄Ａ５７Ａ，Ｌ＄３２將其存在磁碟。

大家可在即期的程式磁碟中找到同一名字的 FILE。

列表(一)

```
1  REM  ::::::::::::::
2  REM  : INIT POKER :
3  REM  :     BY     :
4  REM  : COMPUHOUSE :
5  REM  ::::::::::::::
10  PRINT  CHR$ (4)"BLOAD FAILSAF
    E.INIT"
20  POKE 40266,207: POKE 40267,3:
     REM  DISABLE FP
30  POKE 40268,207: POKE 40269,3:
     REM  DISABLE INT
40  POKE 42336,76: POKE 42337,122
    : POKE 42338,165: REM  DETOU
    R INIT TO NEW ROUTINE
50  PRINT "DONE"
```

程式設計原理：

在ＤＯＳ系統中，ＩＮＩＴ指令是位於＄Ａ５４Ｆ的，此部份的ＣＯＤＥ是儲存了VOLUME NO. 在VOLVAL中，之後再JMP去SETPAR 而完成ＩＮＩＴ工作。另一方面，ＣＡＴＡＬＯＧ指令是位於＄Ａ５６Ｅ的，幾乎立即去到ＳＥＴＰＡＲ而完成ＣＡＴＡＬＯＧ工作！若磁碟是已ＦＯＲＭＡＴ的，則ＣＡＴＡＬＯＧ會正常工作；相反的，若磁碟是未經ＦＯＲＭＡＴ的，則ＣＡＴＡＬＯＧ時會得出DISK I/O ERROR!這個程式就是利用上述關係及原理寫成的！換句話說，當CATALOG得ERROR時即ＩＮＩＴ磁碟；若ＣＡＴＡＬＯＧ正常則顯示之。

好了！現在我們依據ＦＡＩＬＳＡＦＥ．ＩＮＩＴ的ＡＳＳＥＭＢＬＹ ＬＩＳＴＩＮＧ 來解說此程式！

首先，ＬＩＮＥ26及27 是存了VOLUME NO. 使之不能被ＰＡＴＣＨ擦去。ＬＩＮＥ32及33在ＳＥＴＰＡＲ 碼的最後放入一個ＲＴＳ，以避免其JUMP去ＦＩＬＥ ＭＡＮＡＧＥＲ ＤＲＩＶＥＲ。接着，ＣＡＴＡＬＯＧ碼是被放在累加器中而程式會再JMP去SETPAR。而在回來後，累加器（ＡＣ－ＣＵＭＵＬＡＴＯＲ）之內容是會被ＳＡＶＥ起來，而SETPAR後面的暫時性損壞此時是會恢復正常，使之能執行其他DOS指令。

ＬＩＮＥ45 是直接呼叫ＦＩＬＥ ＭＡＮＡＧＥＲ 及執行ＣＡＴＡＬＯＧ工作。若此工作時出現I/OERROR則表示ＤＩＳＫ是未ＦＯＲＭＡＴ的，而ＬＩＮＥ４６～４８便執行ＩＮＩＴ工作。另一方面，若ＣＡＴＡＬＯＧ時有正常反應或其他錯誤，喇叭便會ＢＥＥＰ一聲，並等待用家按鍵（ＬＩＮＥ49，50）。至於ＬＩＮＥ54～57就是恢復VOLUME NO.，繼續ＩＮＩＴ工作！ ■

新 INIT 指令

```
                    1  ************************
                    2  *                      *
                    3  *   FAILSAFE.INIT      *
                    4  *                      *
                    5  * COPYRIGHT (C) 1985   *
                    6  *                      *
                    7  * BY COMPUTING AGE     *
                    8  *                      *
                    9  ************************
                   10  * MERLIN ASSEMBLER
                   11  *
                   12  * THIS A PATCH TO DOS 3.3
                   13  * DISABLES INT AND FP
                   14  * DISPLAYS CATALOG ON INITIALIZED DISK
                   15  *
                   16           OBJ  $A57A
                   17           ORG  $A57A
                   18  VOLVAL   =    $AA66    ;DOS STORE VOLUME NUMBER HERE
                   19  GETCHAR  =    $FD0C    ;MONITOR ROUTINE TO INPUT 1 CHAR.
                   20  FILEMAN  =    $AB06    ;MAIN ENTRY TO FILE MANAGER
                   21  SETPAR   =    $A2AA    ;SET UP PARAMETERS FOR FILE MANAGER
                   22  ERRCODE  =    $B5C5    ;CODE RETURNED TO FILE MANAGER
                   23  BELL     =    $FF3A    ;MONITOR ROUTINE
                   24  SVOLVAL  =    $2F0     ;STORE VOL. # IN KEYBOARD BUFFER
                   25  *
                   26  *
A57A: AD 66 AA     27           LDA  VOLVAL   ;SAVE VOLUME NUMBER
A57D: 8D F0 02     28           STA  SVOLVAL
                   29  *
                   30  * SET FILE PARAMETERS
                   31  * BUT DON'T JUMP TO FILE MANAGER DRIVER
                   32  *
A580: A9 60        33           LDA  #$60     ;OP CODE FOR RTS
A582: 8D E7 A2     34           STA  $A2E7
A585: A9 06        35           LDA  #$06     ;CODE FOR CATALOG
A587: 20 AA A2     36           JSR  SETPAR
                   37  *
                   38  * RESET THE JUMP
                   39  *
A58A: 48           40           PHA
A58B: A9 4C        41           LDA  #$4C     ;OP CODE FOR JUMP
A58D: 8D E7 A2     42           STA  $A2E7
A590: 68           43           PLA
                   44  *
                   45  *
A591: 20 06 AB     46           JSR  FILEMAN  ;ATTEMPT A CATALOG
A594: AD C5 B5     47           LDA  ERRCODE
A597: C9 08        48           CMP  #$08     ;IF WE GET AN I/O ERROR DISK IS CLEAN
A599: F0 06        49           BEQ  OK
A59B: 20 3A FF     50           JSR  BELL     ;WARN USER
A59E: 20 0C FD     51           JSR  GETCHAR  ;CONTINUE ON KEY
                   52  *
                   53  * CONTINUE WITH INIT
                   54  *
A5A1: AD F0 02     55  OK       LDA  SVOLVAL
A5A4: 8D 66 AA     56           STA  VOLVAL
A5A7: A9 9D        57           LDA  #$9D     ;CODE OVER-WRITTEN BY POKES

                        ===== Page 2 =====

A5A9: 4C 63 A5     58           JMP  $A563    ;JUMP TO WHERE WE DETOURED INIT

--End assembly--

50 bytes

Errors: 0
```

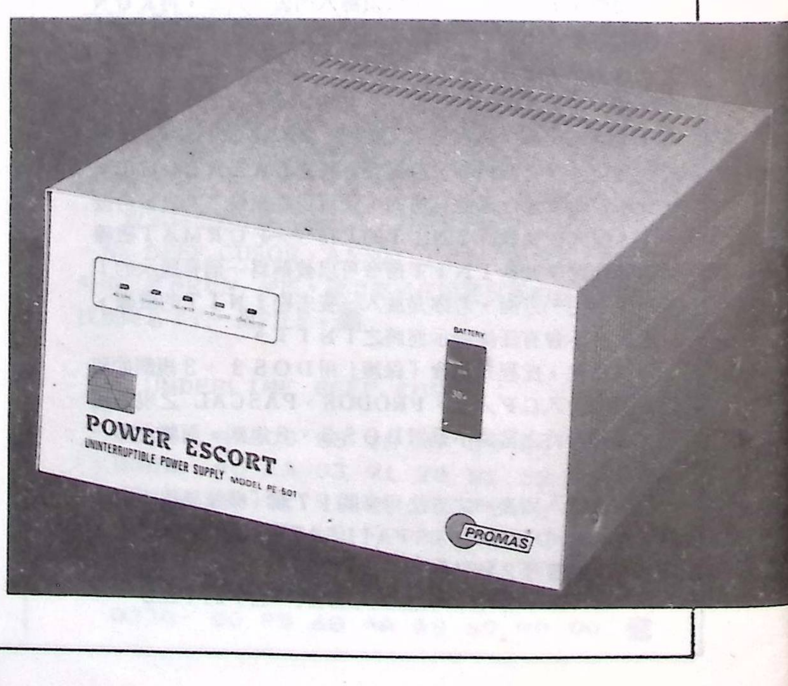

配備後備電源供應器・確保資料安全

電源中斷，電壓驟降等意外，每令閣下痛失寶貴資料、或損壞昂貴精密電腦器材，損失無可挽救！

精明的電腦用家，事事計劃周詳，自當配備PROMAS自動後備供電器，以備電源故障時，隨時提供足夠電源，確保電腦運作可順利完成。這部後備供電器，功能強大，優點包括有：

- 可用作防止電壓下降，電壓冲擊及電源中斷等，確保電源供應正常。
- 當電力停止供應時，能在500分之1 秒內自動提供後備電源，並以蜂鳴器警報及指示燈顯示。
- 內置過荷保護線路及多種過濾器，提供高質素電源。
- 供電時間較一般自動後備供電器爲長。
- 在電力供應恢復正常後，本機隨即自動充電。
- 外型簡單美觀，適用於任何電腦，硬碟儲存系統、精密電子儀器及一般電器使用。
- 不同型號，多種功率，以供選擇（PE－250、PE－501及PE－1000 A）。

PROMAS 自動後備供電器香港總代理是寶馬士電腦有限公司，電話3-674687，歡迎査詢詳情。■



加强控制 JOYSTICK 能力

令 BLAZING PADDLES 更完美

WIKE CHAU

BLAZING PADDLES 爲BAVDVILLE公司提供給APPLE 用家的繪圖工具，BAVDVILLE 公司的繪圖工具一向水準極高，如TAKE ONE 便是一個很好例子。

BLAZING PADDLES可適用於各種控制設備，包括老鼠、光筆、EASY PAD、JOYSTICK 等…，當用家選擇了適當的控制設備後，便可進入BLAZING PADDLES 多姿多采的繪圖世界！在MENU 的版圖裏，BLAZING PAD－DLES的所有功能已呈現在畫面上，而每個功能都有圖形及文字表達它的作用，用家如選擇HELP 則能更清楚其操作細節，使運用時更揮洒自如。

由於有許多功能都是看圖便懂應用，所以筆者並不打算詳述。

筆者最欣賞WINDOW 這個功能（在別的繪圖工具中沒有），它能令用家把畫面中一部份取出，加在畫面其他部份或加在其他圖畫上，甚至可以儲在磁碟上，必要時可從磁碟中取回，放在喜歡的位置，十分好用！

筆者和朋友談及BLAZING PADDLES 時，大家都有同感，BLAZING PADDLES 有一個致命傷，令用家望而却步！那便是使用 JOYSTICK 時游標漂浮不定，使用家控制困難，相信大部份讀者也擁有JOYSTICK ，也使用 JOY－STICK 爲工具，但BLAZING PADDLES 使用 JOYS－TICK 的效果十分差，已到了忍無可忍的地步！筆者有些朋友改用EASY PAD 作工具，效果無疑比 JOYSTICK好，但仍有些點不能繪在畫面上，而且EASY PAD 需要二百多元一套，始終不及 JOYSTICK 普及便宜。因此筆者便起了改良BLAZING PADDLES 的念頭，發覺它游標的定位是根據PADDLE 值直接定位，因此回中 JOYSTICK 便不穩定及有些點不能繪在畫面上（因畫面水平解析度爲280點，而PADDLE 值則爲256），知道了原因後，筆者便思考改良的方法，既然大部份用家都是使用回中JOYSTICK，我們何不就利用回中 JOYSTICK 特性來改良。

經過數小時工作，應用多種工具，包括WILDCARD 、CIA及筆者自己的搜尋工具，終於研究出改良的方法及應更改的位址，方法有二，一是利用 WILD CARD 中斷程式，按"M" 字選擇MONITOR 功能，再如下所示便可成功。

```
C08B C08B
D81D:4C 47 09
D82E:4C 5C 09
D947:C9 F7 B0 07 C9 10 90 08
     4C 25 08 E6 71 4C 25 08
     C6 71 4C 25 08 C9 F7 B0
     07 C9 10 90 08 4C 36 08
     E6 72 4C 36 08 C6 72 4C
     36 08
```

完成之後再按

```
C050 C057 B77G
```

便可成功，現在你便可輕鬆自如地使用BLAZING PADD－LES 的繪圖功能了！但這一方法祇是治標，至下一次 BOOT 碟再使用 BLAZING PADDLE 時便要從新再做；而第二種方法則治本，祇需修改一次以後便不須再修改：

首先BACK UP一隻BLAZING PADDLES ，再以 SECTOR EDITOR 修改TRACK $ 1，SECTOR $ 2。

位址 $1D:4C 47 09

位址 $2E:4C 5C 09

完成之後再修改TRACK $ 1，SECTOR $ 3。

```
$47:C9 F7 B0 07 C9 10 90 08
    4C 25 08 E6 71 4C 25 08
    C6 71 4C 25 08 C9 F7 B0
    07 C9 10 90 08 4C 36 08
    E6 72 4C 36 08 C6 72 4C
    36 08
```

完成之後這片磁碟便已擁有改良功能，以後BOOT起這片磁碟便不需再修改了。 ■

更正

第二十二期「磁碟壽命性能測試器」APPLESOFT 程式列表行句210

原句爲 210 CALL 825

更改爲 210 CALL 823

第廿二期P.86「趨向完美的DOS」右邊一欄PATCH RUN 的修改刊漏了一段，應加上下列一段在AA2C位址之後：

```
A220 :20 A8 A2 20 16 A3
      AD C2 B5 60
```

LOCK IT UP 及APPLEGUARD 所PROTECT出來之磁碟的破解

■ 鄭家輝

LOCK-IT-UP 是一隻非常厲害的磁片保護程式，我甚至用 EDD Ⅲ 都不能拷貝它。我曾經用過前幾期「電腦時代」教過的方法都不能拷貝它，可能是他所說的磁碟機轉速不對。

不過現在我有一種很方便而又很簡單的方法來解拆 LOCK-IT-UP PROTECT 出來的碟（LOCK-IT-UP 本身也是）。

BOOT 起一隻 DOS3.3的碟（快 DOS 也可，例如 DIVERSI DOS，PRONTO DOS 等等）。跟着打入：

CALL-151
B942：18（除去 CHECKSUM）
BAAA：00（更改讀取轉換表）
3D0G（返回 BASIC）

現在各位可以 CATALOG LOCK-IT-UP 的碟，LOAD 入 PROGRAM 也可。各位可以用 FID 等 COPY PROGRAM 來把每一個 PROGRAM COPY 出來，或用 COPYA 來把全隻碟解拆，如果用 COPYA，要把 DOS3.3（或快 DOS）放入前面3條軌。各位有了這個方法，LOCK-IT-UP 便無所頓形了。

APPLEGUARD 是另一個磁片保護程式，它的保護方法與 LOCK-IT-UP 有所不同，它把 TRACK $3 至 TRACK $22 的 ADDRESS MARKER 改變了，又把 TRACK $5清除了，使 NIBBLE COPIER 拷貝它時有錯誤，所以有一些初學者不能拷貝它。

不過現在我有一種方法可以很容易地解拆它，方法如下：

BLOAD ULTRA FASTCOPY CALL-151
809：DA
803G

設定 SOURCE DRIVE 和 TARGET DRIVE，再按 (C)OPY，當 TRACK $0,1,2和5有錯誤時按 N，當 COPY 完了，你便可以把 DOS3.3（或快 DOS，e.g.DIVERSI DOS）放回前3條磁碟軌，便大功告成了。

如果大家想知道那隻碟是不是 LOCK-IT-UP PROTECT 的，大家可以用 NIBBLE EDITOR（如 NIBBLES AWAY Ⅱ 的 TRACK/BIT EDITOR 或 COPY Ⅱ PLUS5.0 的 NIBBLE EDITOR）READ 入 TRACK，如果 ADDRESS MARK（D5AA 96）之前是 EB，那多數便是 LOCK-IT-UP PROTECT 的。而且 LOCK-IT-UP PROTECT 出來的碟多數是要 WRITE PROTECT 的。

而 APPLE GUARD 所 PROTECT 出來的碟是把 ADDRESS MARK 改成 DA AA 96，所以大家便要用 NIBBLE EDITOR 看看，如果 TRACK $3- $22（TRACK $5除外）的 ADDRESS MARK 是 DA AA 96，和 TRACK $5全都是 FF，那多數便是 APPLE GUARD PROTECT 出來的。■

拆LOCKSMITH 5.0 16 SECTOR FAST COPY 爲FILE

■ NIBBLE

相信大家用過LOCKSMITH 5·0後，都會發覺到裏面的16 SECTOR FAST COPY 十分具有威力。這個COPY程式不但有驚人速度，在單磁碟機COPY 時，需要轉碟的次數也是最少的。但是每次使用前都要將整張LOCKSMITH BOOT 起，十分麻煩！所以筆者提供一個將它拆成FILE的方法。

工具：16K RAMCARD 和LOCKSMITH 5·0一張

方法：

1. 將RAMCARD 插在SLOT 0
2. 打開電源後立刻按＜CTRL－RESET＞
3. CALL－151
4. C081 C081
5. D000＜D000·FFFFM
6. FE35：4C 03 E0
7. C080
8. 按＜CTRL－C＞回到BASIC
9. BOOT LOCKSMITH 5·0
10. 按＜SPACE＞到MENU
11. 按U
12. 按B
13. 按＜SPACE＞
14. 待程式要求你入ORIGINAL時按2次＜RETURN＞
15. 此時應該回到BASIC 狀態
16. CALL－151
17. C082
18. 1672：02
19. 177E：EA EA
20. 7FD：20 00 20
21. 2000：A0 10 A9 FF 20 A8 FC 88 10 F8 60
22. 4000＜7FD·200AM
23. 按＜CTRL－C＞回到BASIC
24. BOOT一隻正常DOS 的磁碟，如：DOS 3·3
25. 回到BASIC 狀態
26. CALL－151
27. 7FD＜4000·5810M
28. 按＜CTRL－C＞回到BASIC
29. BSAVE FILENAME，A$7FD，L$1810

成功！ ■

徵求一行及兩行程式設計有獎遊戲

自從徵求一行程式遊戲舉辦以來，獲得許多讀者的支持，每月均收到數以百計的作品，當然，部份有水準的作品我們經選登出來，至於未被接納的作者請勿灰心，繼續努力，多些編寫。

由於利用一行敍述句來寫出一個精彩的程式有一定的難度限制，未必能充份發揮個人的創作才華，爲此，我們再舉辦多一個遊戲——徵求二行程式設計，讓大家有二行敍述句的長度來發揮個人的才華。

二行程式設計比賽參加辦法及獎品和一行程式有獎遊戲一樣，但請在信封面註明二行程式。

各位讀者寫來程式時，必須用多行的形式寫來（參考本文的各位得獎例子），而且要書寫清楚。投稿請寄來CPO BOX 71193 。入選者可獲得任何一期的程式磁碟一張（請註明要的磁碟）。來稿信封請勿封口。 ■

本程式是第十六期的一行程式的改良版本，是一個眞正的COPY PROGR－AM，名爲MINI COPYA。

本程式可COPY PRODOS，CPM，DOS 3·3 等等，總之凡是COPY A 做到的本程式也能做到，基本原理和COPYA一樣，但因爲使用BASIC LANGLA GUE的關係，時間比COP－YA 慢，但是若遇到READ ERROR的話，程式將繼續（類似LOCKSMIT－H 5·0 的FAST COPY）。

若要COPY 40 TRACK的磁碟，只要將20行的所有“34”改爲“39”則可。

COPY完成後兩部磁碟機都停，表示已完成，請按RESET 鍵後再BOOT機。

註：1.請不要改變輸入的次序。

2.本程式佔用＄2000至＄A000 做BUFFER，所有該區的記憶將被清洗。

李荔洲同學可取得第廿三期程式磁碟一張。■

```
10    HOME
    : PRINT "MINI COPYA,CREATED ON 14/1
      0/1985"
    : PRINT "IN PUI YING MIDDLE SCHOOL"

    : PRINT
    : INPUT "PRESS<RET> TO START";A$
    : PRINT CHR$ (4)"INITT,D2"
    : A = 47092
    : C = 768
    : POKE A - 1,0
    : POKE A - 9,0
    : FOR I = 0 TO 5
    : READ T
    : POKE C + I,T
    : NEXT
    : FOR K = 1 TO 10
    : FOR F = 1 TO 2
    : DATA 32,227,3,76,217,3
20    FOR I = 8191 TO 40959 STEP 256
    : B = A - 8
    : D = B + 1
    : E = E + (S ) 15)
    : S = S * (S ( = 15)
    : L = INT (I / 256)
    : POKE A - 4,I - L * 256
    : POKE A - 3,L
    : POKE D,S
    : POKE B,E
    : POKE A,F
    : POKE B - 2,F
    : K = (E = 34) * 10
    : CALL C
    : S = S + 1
    : I = (E = 34 AND S = 16) * 40959
        + I
    : NEXT I
    : E = G * (F = 1) + (F = 2) * E
    : G = E
    : S = (F = 1) * H + (F = 2) * S
    : H = S
    : NEXT F,K
```

本程式可將PRINTSHOP 的GRA－PHIC 印出高解像第一頁。不過程式執行時，要耐心等待一段時間。余衍焯同學可贏得他希望得到的一期程式磁碟。■

```
10    INPUT "FILENAME:";F$
    : PRINT CHR$ (4)"BLOAD"F$
    : HGR
    : A = 22528
    : FOR Y = 1 TO 52
    : FOR X = 1 TO 11
    : C = PEEK (A)
    : FOR B = 0 TO 7
    : V = 2 ^ (7 - B)
    : HCOLOR= ((C = ) V) * 3)
    : HPLOT X * 8 + B,Y
    : C = C - (C = ) V) * V
    : A = A + .125
    : NEXT B,X,Y
```

這是一個簡單而考眼力的射擊遊戲。執行後，首先要輸入射擊次數及速度。速度中300最慢，而 1 最快。當 "；" 出現後，立即記下它的位置及鍵入適當的數字，如果偶一射不中，程式便會結束。鍵入程式時不能有空位鍵入。

洪力沛同學可到來讀者服務部領取第22期程式磁碟一張。■

```
10    HOME
    : INPUT "TIMES(1-99)";M
    : INPUT "SPEED(1-300)";N
    : L = 0
    : FOR G = 1 TO M
    : A = INT ( RND (1) * 9) + 1
    : V = 15
    : VTAB V
    : HTAB V
    : PRINT "::::::::::"
    : VTAB V
    : HTAB V + A
    : PRINT ";"
    : FOR W = 1 TO N
    : NEXT W
    : HOME
    : VTAB V
    : HTAB V
    : PRINT "0123456789"
    : INPUT C
    : IF A = C THEN L = L + 20
    : NEXT G
    : PRINT "SCORE=";L
```

這是一個地心探險的遊戲，玩者的飛船深入崎嶇的地層，若碰到山石便算輸。程式中把低解像繪圖的指令應用在文字幕上，做出畫面效果及測試有否碰到山石。

李應虹君可取得第14B 磁碟一張■

```
10    W = (W = 0) * 10 + W - .01 + (W
      ( 3)
    : K = PEEK (49152)
    : X = X - (K = 136) + (K = 149)
        + (X = 0) * 10
    : L = (L ( 4) * 2 + L + SGN (
      RND (1) - .5) - (L + W ) 30)
    : COLOR= 12
    : HLIN 0,L - 1 AT 47
    : HLIN L + W + 1,34 AT 47
    : C = SCRN( X - 1,21)
    : HTAB X
    : VTAB 11
    : PRINT "V";
    : HTAB 37
    : VTAB 24
    : T = T + 1
    : PRINT
    : IF C ( ) 12 THEN 10
```

利用我們的分類廣告，閣下只要花極少的代價，便可將訊息傳到我們廣大的讀者網

無論是自己編寫的軟件、轉讓各類二手電腦及週邊、新出版書刊、代編程序服務、私人電腦科補習、程序員徵聘、待聘、軟件出讓等等，都可以利用電腦時代的分類廣告欄。

查詢詳情，請電：3-312007分類廣告部洽

黃松明書籍軟件郵購

(1) APPLE CP／M操作法一HK$ 11
(2) 蘋果軟件集錦一HK$ 13
(3) APPLE CP／M軟件集錦一HK$ 14
(4) GAME I／O 控制應用一HK$ 8
(5) 印字機控制程序一HK$ 10

書籍和軟件目錄函索即寄，香港地區免郵資。電3-7283169
科德技術服務公司香港九龍長沙灣道266號昌發大厦六樓六號室

徵求彩色監視器

誠徵任何牌子彩色監視器要有聲响使用期間不超過兩年爲佳，出價六百元左右有意電0-6491147華仔洽。

PRINTSHOP用多款造型加菲貓

十五款不同造型加菲貓專輯，專供PRINTSHOP使用，可供設計印製各種問候咭、生日咭、海報或橫額等。磁碟每張十五元。電腦時代讀者服務部出售，只限長期訂戶／AFC／MBC會員。可郵購。

港人創作ADV遊戲

港人創作ADVENTURE遊戲軟件——地鐵風雲，由專欄作者李德華君創作，每個只售15元。本軟件並未加上任何保護，可隨意LIST。只售與長期訂戶、AFC會員及MBC會員。讀者服務部發售。

CP/M用戶佳音

CP／M用戶佳音，CP／M FORMAT 40（VERSION II），將CP／M格式磁碟FORMAT成40軌，每套50元連使用手冊。電腦時代讀者服務部發售。

精工修理電腦／週邊咭

精工修理蘋果兼容及IBM兼容電腦及週邊咭，取價公道，深水埗黃金商場地庫21號萬達電腦公司。3-610095

TAKE I 製作電影

TAKE 1 電影製作比賽得獎作品——約會，每張十五元（連磁碟）電腦時代讀者服務部發售。

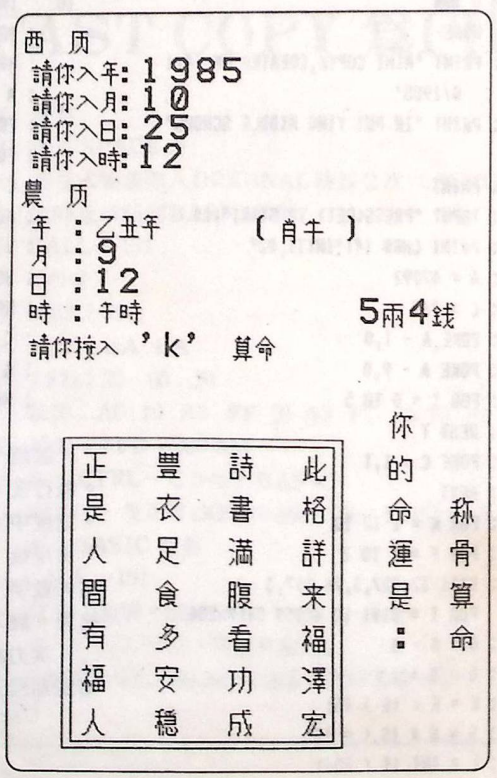
軟體交換蘋果與IBM

本人收集蘋果與PC軟體各千餘種欲與人交換，目錄待索，來函請寄台灣台北市基隆路1段83巷40號2F楊忠明。

稱骨算命程式

中文稱骨算命程式，港人編寫，根據出生年、月、日、時計算個人一生運程，非常準確，每套25元，只售與MBC會員及本刊長期訂戶及AFC會員。發售處：電腦時代讀者服務部。歡迎參觀示範。

無須中文咭，可出印字機，只須輸入陽曆日子即可，免查農曆時辰八字。

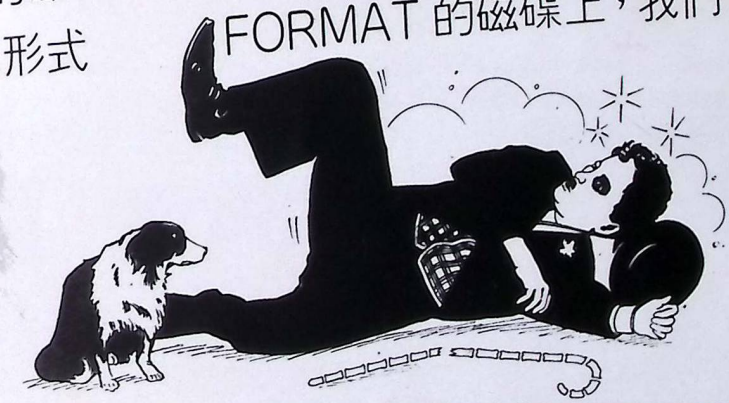


閣下有否因添換IBM系統，却爲面對原本儲存的APPLE機FORMAT舊資料而煩惱?!

我們有辦法幫到你 !!

IBM ⇄ APPLE

任何形式的磁碟資料，我們都有辦法將它轉換上IBM機FORMAT的磁碟上；相反，將IBM FORMAT的磁碟轉換於任何形式FORMAT的磁碟上，我們也有辦法爲閣下解決。



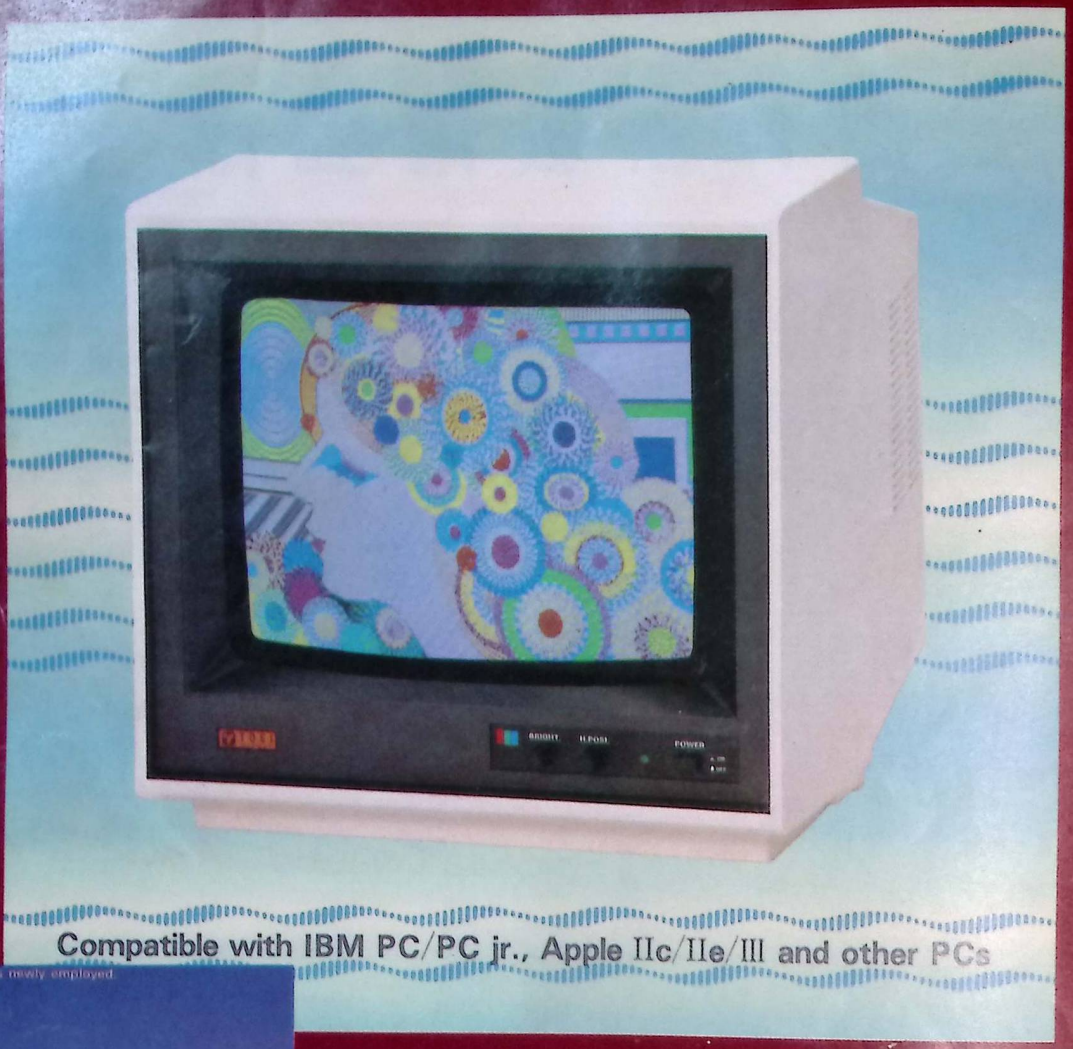
任何疑難，我們都會有專人爲閣下解答。

請於每日下午三時至六時，撥電3-312007
電腦時代出版社，伍先生洽

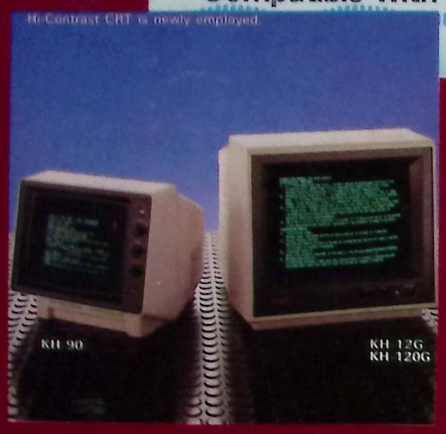
TOEI Toei Electronics Co., Ltd.

TOEI 系列彩色監視器

色彩更趨完美・影像更見逼眞



Compatible with IBM PC/PC jr., Apple IIc/IIe/III and other PCs



FTC系統規格表

Name of Model		FTC-1201-R	FTC-1203-R	FTC-1208-R	FTC-1203-HF	FTC-1208-HF	FTC-1208-W
Picture Tube		12", 90°Deflection, In-Line Type, P22, Hi-Contrast CRT (FTC-1203-R, FTC-1203-HF : Non-Glare)					
		0.47mm Dots Grade Slope Pitch	0.38mm Dot Trio Pitch	0.28mm Dot Trio Pitch	0.38mm Dot Trio Pitch	0.28mm Dot Trio Pitch	
Input Signal Level	RGB Video Signal	RGB Analog 2~5Vp-p (Positve), RGB plus Intensity TTL Level (Positive) or RGB TTL Level (Positive)			RGB plus Intensity TTL Level (Positive)		
	Sync. Signal	Composite or Separate (H. D., V.D.) TTL Level (Positive or Negative)					
Scanning Frequency		H. 15.75kHz ±400Hz, V. 60Hz (50~65Hz)			H. 25kHz ±600Hz, V. 60Hz (50~65Hz)		H. 31.5kHz ±1kHz V. 60Hz (50~65Hz)
Video Bandwidth		20MHz-3dB			25MHz-3dB		
Characters to Display		2,000 Characters (80×25 Lines)			4,050 Characters (90×45 Lines) or 2,000 Characters (80×25 Lines)		7,200 Characters (120×60 Lines) (Ex.)
Active Display Area		H. 220mm (Max. 240mm), V. 137mm (Max. 150mm)					
Active Display Time		H. 45μsec., V. 13.1msec.			H. 30.2μsec., V. 13.1msec.		H. 23.8μsec. V. 13.1msec.
Resolution		640Dots (FTC-1208-R : Max. 800Dots) ×200Lines (Max. 280Lines)			640 Dots ×400 Lines (Max. 480 Lines)	640 Dots (Max. 800 Dots) ×400 Lines (Max. 480 Lines)	912 Dots (Max. 800 Dots) ×912 Lines (Max. 950 Lines)
Amplifier Type		Linear Amplifier					
Connector Type		8P Terminal					
Power Supply		AC 120V (85~132V) AC 220V (180~260V) 60Hz (50~60Hz)					
Power Consumption		Approx. 70W			Approx. 80W		
Environment	Operating Temperature	0°C~+50°C					
	Storage Temperature	-10°C~+65°C					
	Storage Humidity	20%~80% (Non-Dew)					
Weight		Approx. 10kg net					
External Size		324 (W) ×295 (H) ×367 (D) mm					
Size of Packing	Single carton box	387 (W) ×456 (H) ×397 (D) mm : 12kg					
	Double carton box	459 (W) ×527 (H) ×476 (D) mm : 14kg					
FCC		Class B approved			Class A approved Class B complied		

TOEI Toei Electronics Co., Ltd.

特約經銷商

萬達電腦公司

深水埗黃金商場地庫21號

電話：3-610095

傳真：3-610082

* IBM PC、PC/XT is registered trademark of International Business Machine Corp.

** Apple II is a registered trademark of Apple Computer Inc.

蘋果工具箱

52 ■ 程式保留字檢查系統
[illegible]
[illegible]

BACK UP資料館

73 如何抄錄U4
79 ■ 專爲拷貝BROTHERBUND軟件而設的POWER COPY
82 把SUMMER GAME II拆爲COPY A版本
88 ■ 拆LOCKSMITH 5.0 16SECTOR FAST COPY爲FILE
88 LOCK IT UP及APPLEGUARD所PROTECT出來的磁碟之破解法

最新遊戲軟件評介

69 ULTIMA IV的玩法 撰稿：先鋒電腦軟件公司
74 U4與U3的比較 再談BUCKHOO BANZAI
RESCUE ON FRACTALUS WINTER GAMES

精品店

73 第五代電腦也可享用MCS
83 ■ 配上CLOCK咭應用的私人約會備忘日曆
84 FORMATING磁碟雜談
87 加强控制JOYSTICK令BLAZING PADDLE更生色

讀者園地

13 電腦時代技術叢書出版介紹
48 各期程式磁碟總目錄
51 電腦時代精裝合訂本
78 訂閱表格
75 AFC通訊
87 更正
89 一行兩行程式參賽入選作品
90 分類廣告

編者言

第廿四期新加入的dBASE II 應用技巧專欄文章：發票資料檔案系統的設計，同時出版一張獨立的磁碟版本發售，每張只售15元，磁碟版本亦同時提供有APPLESOFT CP／M FORMAT及IBM MSDOS FORMAT。讀者服務部發售。

麥法基君今期爲電腦時代編寫的遊戲創作專欄中終一個冒險遊戲TRANSYLVINA，改編並配上特殊音响效果供每有MOCKINGBOARD的讀友享用，新改編的遊戲有獨立一張磁碟發售，每張15元，只限MBC、AFC會員及電腦時代長期訂戶。連同程式磁碟一併訂閱的長期訂戶可免費獲得COPY，請携磁碟到讀者服務部。 ■

第 24 期電腦時代程式磁碟

可於下列地點購得：

(1) 各地鐵／九廣鐵路香港電視服務站；
(2) 百老滙攝影器材有限公司港九各店；
(3) 各區指定經銷處；
(4) 電腦時代讀者服務部

每張港幣29.90元

特稿

■ Program saved on disk

■ II＋上的雙倍高密度解像 24
程式編寫入門之四：EXEC指令的用途 43
新讀者需知 44
新增加兩項有獎遊戲 PS圖案設計/MCS作曲大賽 46

中文咭應用專欄

■ 改良版中文大字海報編印器 23

應用軟件庫

■ 趨吉避凶的生理周期分析預報系統 26

dBASE II應用專欄 弘智電腦智識中心供稿

發揮dBASE II最大優點的發票資料檔案系統 15

小型製作專欄

SUPER ROM／F8合二爲一體的 DUAL ROM 81

齊齊樂開心坊

怎樣把MKB的說話部份加進自己的程式內 41
配有特殊音响效果的TRANSYLVINA 42
■ 棋類遊戲百種之八 戰略跳棋TACTICS 60
■ 被困在迷宮的小猩猩 64

軟件縱橫談

文字及圖案編輯處理器：FONTRIX 30

專欄

卡通人物專門店 查理與史諾比 32
■ 麥法基專欄 起始地址搬移器 35
機械語言應用 圖形繪畫法研究(四) 38
PRODOS縱橫 滙編語言編譯器的使用方法 68

兒童電腦輔助教育

■ 方向概念訓練：東南西北 57

學校電腦輔助教育

現年度蘋果教育基金開始接受各界申請 76

電腦時代

本刊所登載之各個程式全部適用於APPLE系列或兼容機種電腦之上

NO. 24

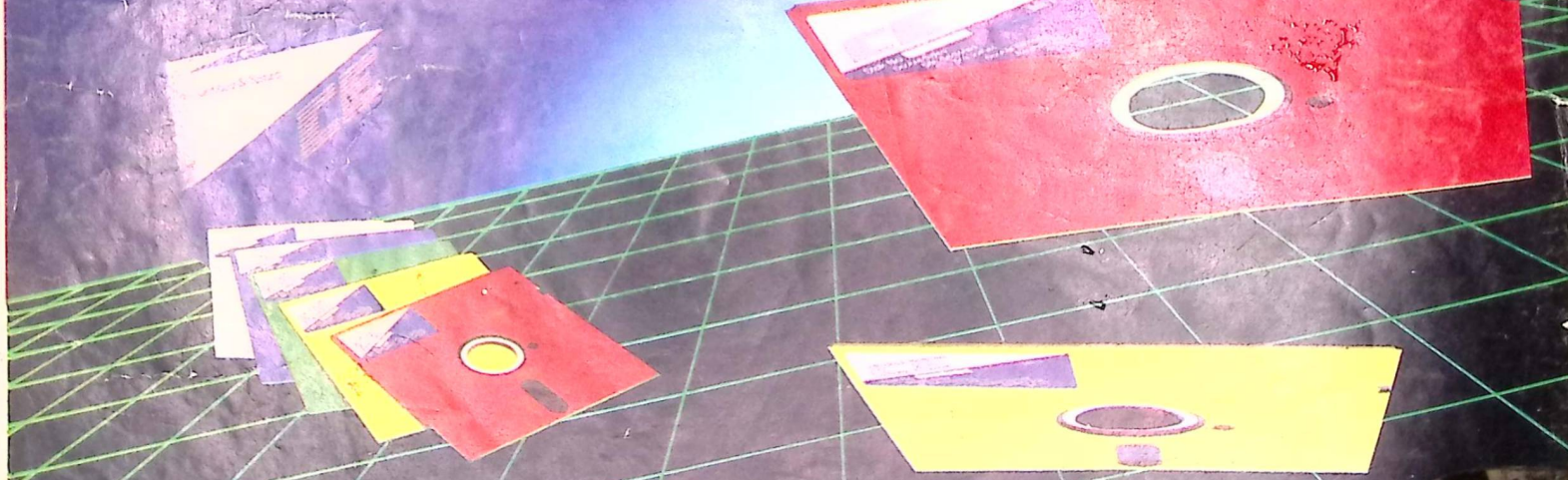
本刊每月廿五號出版。一年十二期。全年訂閱費120港元，雜誌連程式磁碟爲475港元。海外訂戶請參看訂閱收費表。

■出版：電腦時代出版社 ■督印人：伍小華 ■編輯：電腦時代編輯部 ■軟件支援：COMPUHOUSE ■通訊地址：九龍中央郵箱71193號 ■通訊電話：3-312007 ■ 讀者服務部地址：旺角上海街438號同珍商業中心11字樓1104室（登打士街交界）
■承印者：永恒印刷公司 ■發行人：同德書報社

CE Disk

彩色磁碟 電腦知己

可重覆寫讀100,000,000次以上



● ＣＥ彩色電腦磁碟，是當代最先進的產品，令人在單調的電腦操作中，耳目一新，平添不少彩色繽紛的生氣，誠爲電腦用家的知己良伴。

● ＣＥ彩色電腦磁碟特有的彩色保護套，不留指印，既能防塵侵染，又可免磨擦生電。除顏色鮮艷外，更易於分門別類，以適合各用家的不同須要。

● ＣＥ彩色電腦磁碟表層磁性幼細平滑，並經特製之潤滑塗膜處理，可抵禦長期的磨擦，使碟內的資料得以保存無損。在適當的操作下可承受100,000,000次以上的重覆寫讀。

● ＣＥ彩色電腦磁碟比其他產品更具抗熱和防震功能，不會出現彎曲和摺疊現象，可說是「全天候」磁碟。保證在正常環境下操作，寫入資料與讀出時100％無誤差。

香港代理：華益企業公司
NO.31-33 KING'S ROAD, 1ST FLOOR,
(NEAR LAU LI ST.) NORTH POINT.
G.P.O. BOX 10617,
HONG KONG

TELEX NO: 75632 PECO HX
CABLE ADDRESS: "PETCOMY" HONG KONG
TELEPHONE: 5-784928, 5-783129
BANKER: HANG LUNG BANK LTD.